青少年计算机学习与应用丛书

苹果机和中华学习机
应用实验与实用制作

胡礼和　周卓伦　编著

清华大学出版社





苹果机和中华学习机应用实验与实用制作

胡礼和　周卓伦　编著

清 华 大 学 出 版 社

内 容 简 介

本书主要介绍在APPLE-Ⅱ微型机（中华学习机的兼 容机，兼顾其它常见微型机）上的一些简单易行的应用实验、实用制作和使用技巧，以及常见的故障分析和排除方法。其中包括23个很有趣味的上机小实验和16个有一定使用价值的小制作。

本书可供青少年学习计算机软硬件知识使用。也可供有关教师及科技人员参考。

苹果机和中华学习机应用实验与实用制作

胡礼和　周卓伦　编著

*

清华大学出版社出版

北京　清华园

北京京辉印刷厂印刷

新华书店总店科技发行所发行

*

开本：787×1092 1/32　印张：7 3/8　字数：165千字

1990年 5 月第1版　1990年 5 月第1次印刷

印数：0001～8000

ISBN 7-302-00458-7/TP·158

定价：2.80 元

序

经党中央、国务院批准实施的“星火计划”，其目的是把科学技术引向农村，以振兴农村经济，促进农村经济结构的改革，意义深远。

实施“星火计划”的目标之一是，在农村知识青年中培训一批技术骨干和乡镇企业骨干，使之掌握一、二门先进的适用技术或基本的乡镇企业管理知识。为此，亟需出版《“星火计划”丛书》，以保证教学质量。

中国出版工作者协会科技出版工作委员会主动提出愿意组织全国各科技出版社共同协作出版《“星火计划”丛书》，为“星火计划”服务。据此，国家科委决定委托中国出版工作者协会科技出版工作委员会组织出版《全国“星火计划”丛书》，并要求出版物科学性、针对性强，覆盖面广，理论联系实际，文字通俗易懂。

愿《全国“星火计划”丛书》的出版能促进科技的“星火”在广大农村逐渐形成“燎原”之势。同时，我们也希望广大读者对《全国“星火计划”丛书》的不足之处乃至缺点、错误提出批评和建议，以便不断改进提高。

《全国“星火计划”丛书》编委会

1987年4月28日

《全国“星火计划”丛书》编委会

序　　言

当前，世界正面临着一场新的技术革命，为了适应新技术革命的发展，我国正在大力开展普及与应用计算机技术。

目前，世界上许多国家的计算机教育的重点已从高等院校转向普通教育、职业教育，计算机正逐步形成普及的趋势。为使我国在十年或十五年以后走上工作岗位的亿万中小学生成为掌握信息社会的工具、具有计算机基础知识的科技人才，国家科委、国家教委、中国科协、电子部等单位联合组织开发了“中华学习机”，以适合中小学教学及家庭使用。这是当前世界新技术革命和教育革命的一大趋势。计算机进入学校，走向家庭，将有力地促进整个社会的进步和发展。

为了适应广大青少年学习、应用计算机的需要，我们编撰了这套“青少年计算机学习与应用”丛书。丛书以“中华学习机”系列微机为背景，除了通俗、简明地介绍计算机的使用、应用以外，还形象生动、深入浅出地介绍计算机原理、软硬件基础知识及应用发展等方面的内容。为了促进中小学生德智体美全面发展，丛书还介绍一系列辅助教育软件，其中有辅导语文、外语、数学等基础课程的学习软件，有开发青少年智力的游戏软件，还有提高文艺修养方面的艺术软件。

希望这套丛书能成为广大青少年的朋友，同时也希望它成为广大在职干部和职工的有益的参考读物。

中国计算机学会　吴几康　陈树楷
全国中学计算机教育研究中心　吕传兴　潘懋德
1987.8

《青少年计算机学习与应用》
丛 书 编 委 会

前　　言

为了使广大的青少年计算机爱好者在学习了一定的计算机软硬件知识的基础上，通过动手实验，进一步学习接口电路和控制技术，培养使用计算机的技能，发展应用能力，增长创造才干，本书介绍了一些简单易行的计算机应用实验和使用技巧，以及有关的实用制作。

考虑到当前国内的实际条件，本书所介绍的实验以APPLE-Ⅱ微型机（中华学习机的兼容机）为主，兼顾其它常见微型机；均未采用成本较高、电路复杂的A/D、D/A接口卡，充分利用了游戏接口的潜在功能；自制的接口电路也尽可能地加以简化；有关程序一般用BASIC语言编写。通过实验，还可制得一些实用装置，这些装置的制作费用不高，却有一定的使用价值。为了方便读者实验，本书还介绍了实验用电源及某些辅助检测装置的自制方法。书中还提供了有关资料，使缺乏实验仪器的读者，仍可完成书中的大部分实验和制作。

只要认真地按本书的要求进行实验和制作，是不会损坏微型机的。如果万一不慎或因其它原因，使机器出现故障，读者还可根据本书介绍的常见故障分析和排除方法，动手试一试。

正如中学物理、化学课程中必须配有一定课时的实验课一样，计算机课程也应开设实验课。学习计算机还需上机实习。本书可作为中学计算机实验课和上机实习的学习材料，

也可供教师教学参考。对于其他计算机爱好者来说，本书提供的资料如能起到抛砖引玉的作用，将使编著者感到欣慰。

本书在编写中参考了叶平等同志提供的材料，全书由张开析高级工程师认真审订，谨在此一并致谢。

编著者

1988.4.

目　录

第一章　电子计算机的应用与自动控制

第一节　计算机的广泛应用

电子计算机简称计算机，是现代科学技术的一项卓越成就，是二十世纪技术革命的一个重要标志，它已经渗透到国民经济和日常生活的各个领域，推动了社会的发展。概括起来，计算机的应用主要有数值计算、数据处理、自动控制和人工智能等方面。

应用计算机进行数值计算的例子是很多的。在计算机出现以前，人们虽然从理论上找到了解决某些问题的计算公式，然而，由于计算太繁杂，实际上无法应用。例如，人们早就知道可以用一组方程来推算天气的变化，但是，用这种公式预报24小时以内的天气，如果由一个人手工计算，将需要几十年，这显然失去了预报的意义。而用一台小型计算机，只需十分钟就能算出一个地区三天以内的天气预报。又如，1965年我国在世界上第一次合成人工胰岛素，以后又测定了胰岛素的分子结构，也是利用计算机对实验数据进行理论上的数值计算才得以完成的。

所谓数据处理就是数据综合分析，是指对科学研究、生产实践、经济活动、甚至日常生活领域中所获得的大量数据，按不同的要求进行各种加工，包括搜集、转换、分类、

组织、计值、存储、维护等等，有的还需绘制数据分布曲线或印出报表。这些处理一般不涉及复杂的数学问题，但是数据量大，时间要求紧迫，只有用计算机才能很快地完成。例如我国第三次人口普查就是使用计算机进行汇总的。如果单靠手工汇总，则需要几十年的时间。在各种管理方面，也都需要使用计算机进行数据处理。

计算机在工业方面最先用于流程控制。例如，一座乙烯合成氨厂有几百个控制阀门和上千个变量需要测量和控制，参数变化从几秒钟到几小时，如果采用人工控制，是很难实现生产过程自动化的。七十年代的一台带钢热轧机，采用计算机控制后，产量提高了一百倍，而且质量显著提高。现在，计算机自动控制不仅广泛地应用于生产过程，而且应用于交通运输、国防建设等各方面。例如，用计算机控制并调度汽车、飞机，可以大大减少意外事故；装有计算机的导弹能自动跟踪目标，选择最佳起爆点……

人工智能主要是用计算机来模拟人的某些智力活动。这方面的研究和应用虽然才刚刚开始，但在机器人、医疗诊断专家系统、定理证明、语言翻译等方面，已经有了较显著的成效。

以上介绍的仅是计算机应用的几个主要方面。随着计算机技术的发展，计算机的应用领域仍在不断扩展，可以说，明天将是计算机的世界。

近年来，微型计算机的发展很快。高档微型机的功能可与小型计算机媲美。微型机还具有体积小、灵活性大、价格便宜等优点，因此，微型机不仅可用于生产、交通、军事，还可进入办公室甚至家庭。在信息社会里，计算机是人们获取信息、处理信息的常用工具。将来，计算机特别是微型机

将会象钢笔、电话一样与每个人打交道。所以，学会使用和应用计算机是现代社会对每一个人提出的要求。

第二节 计算机自动控制

计算机自动控制是一个和国民经济、人民生活密切相关且极有潜力的应用领域，因此，了解计算机自动控制的原理和过程是有必要的。

工业生产上的自动控制，早在电子计算机出现之前就已实现了。早期的自动化采用机械、电力或液压传动等方式，在一定程度上解决了单机和单一生产线的自动化问题，但是随着机器运转和生产过程速度的不断提高，已经满足不了实际的需要。计算机应用于自动控制体系，使自动化的程度提到一个新的高度。

按照生产工艺过程的特点，生产过程的自动控制可分为连续性和非连续性两大类型。

关于非连续性生产，下面以自动机床为例加以说明。

在计算机出现以前，自动机床进行自动控制的主要部件是靠模拟装置或仿形装置。按要加工的形状做成模子，用刀具照模子来加工，仅适用于大批量生产同一种形状的工件。

为了提高机床的自动化程度，人们后来采用了程序控制，就是按照加工的要求编制好刀具运动的步骤，把步骤记在穿孔纸带、穿孔卡片、插销板或磁带上，加工时按照事先规定的步骤发出各种电信号，通过继电器等机电控制装置，传给相应的执行机构，让刀具按步骤自动切削。这一控制原理和电子计算机控制类似。这种程序控制一般采用数字、数据形式，所以这种机床也叫数控机床。

应用计算机作控制装置的数控机床，在加工工件以前，只要给出加工表面的一些关键点的坐标位置以及加工表面几何外形的数学公式，根据这些编制出加工程序，将加工程序输入到计算机内，计算机就能自动地算出刀具的运动轨迹，控制刀具运动。应用计算机的数控机床适应性强，中、小批生产也完全适用。一台计算机可同时控制几台甚至几十台数控机床。它能提高加工精度、确保质量、减轻劳动强度、提高劳动效率。

关于连续性生产，下面以化工产品的生产为例加以说明。

在化工生产中，原料连续不断地送入进料口，通过管道流经一系列工艺设备，最后制成产品，又连续不断地从出料口送出来。其工艺过程与温度、压力、流量、液位、化学成份等理化参数有关。对这一类连续性生产过程实现自动控制，需要自动调节这些物理、化学参数，使生产过程处于最佳状态，实现最佳控制。

应用计算机实现最佳控制，首先要随时掌握实际生产过程中上述理化参数的具体数据，把它们的变化通过变送器转换成电流、电压等电量的变化，这种连续变化的电量叫做理化参数的模拟量。再把这些模拟量通过模拟/数字（即A/D）转换器转换成数字，输入计算机。

要计算机能够判断这些数据是否符合最佳控制，还应先建立影响生产过程的各种参数变化的数学模型，即把生产过程中各种参数之间的关系以及这些参数对产量、质量、原材料消耗等的影响，用数学方式定量地描述出来，然后通过反复地试验分析，确定最佳控制所要求的各种条件，作为电子计算机进行最佳控制的依据。计算机根据每次输入的实测数

据，经过计算、分析，确定应采取怎样的调整步骤，然后发出电信号，通知执行机构进行实际调整。

计算机输出的结果是数字，因此，还要通过数字/模拟（即D/A）转换器转换成电量，再由变送器把电量改变成各种参数量，从而控制生产过程。

上面介绍的控制化工产品的生产过程是由计算机自动接受生产过程中测量的各种数据，计算出控制方案，再返回直接指挥控制部件，通过执行机构去控制生产过程，从而构成一个闭合环路。这种控制系统叫做“闭环控制”。在闭环控制中，操作人员对整个自动控制系统只是给出控制要求，监督系统正常工作，并不直接参与控制，因此，这是一种自动化程度较高的控制方式，属于全自动控制。

在控制系统中，如果计算机仅接受生产过程的测量数据，计算出控制方案，提供给操作人员参考，然后由操作人员决定控制方案，完成控制动作；或者先由操作人员输入测量数据，计算机仅按操作人员的要求去计算出控制方案，控制生产过程，那么，在整个控制系统中起码有一个环节是敞开的，这种控制系统叫做“开环控制”。开环控制系统需要操作人员直接参与，属于半自动控制。

不论是哪一类控制系统，都是应用计算机与其它机器、设备、仪器相联结，对它们的工作实施控制。

第三节　计算机的控制机能

计算机之所以能应用于自动控制，并使自动控制提到一个新的高度，主要是因为它具有一般机器所没有的特点，这就是它的“记忆”能力和判断能力。

计算机能将有用的信息存储起来，需要时再取出来用，就象人类平时把所得到的许多印象记在脑子里，需要时重现出来一样。计算机“记忆”的存储量非常大。一台小型计算机的内存储器可存放几万个数据，外存储器可存放上百万个数据。现代大型计算机的“记忆”能力更是惊人。

计算机里设计有一定的电路。通过类似数值计算的一定运算程序，根据不同的数据和条件，作出判断。判断是人类思考的基本部分，平时我们说的“辨别是非”就是一种判断。计算机靠电路对“开”和“关”作出的两种不同的反应，来决定下一步该怎么办，这就是计算机的判断。虽然这是一种简单的判断，但是综合众多的这样的判断，可以作出十分复杂的判断。就象钢琴上的每个键虽然只能发出一个单音，但是演奏家却能根据乐谱在这样的钢琴上弹奏出千百种优美动听的乐曲一样。

从前面介绍的计算机自动控制原理可以看出，计算机所具有的“记忆”能力和判断能力是它具有控制机能的基础。

实现自动化的重要手段是及时地搜集检测数据，按最佳值进行自动控制或自动调节控制对象，这就是所谓的“实时控制”。计算机之所以能进行实时控制，除了因为它具有“记忆”和判断能力之外，还因为它里面有一个电子时钟。这时钟是由电子逻辑线路构成的，比日常生活中的钟表精确几十倍。计算机还能发出电信号，就是一串串瞬时变化的电流和电压的脉冲。为了控制整个系统的“步伐”，衡量脉冲的变化，计算机里还有一个脉冲发生器，这样便可在固定间隔时间里发出脉冲，从而有了计时的基础。计算机实时控制就是先将计算机的时钟与当地标准时间取得一致，然后根据实际需要来实现对各种仪器设备的控制过程。

自动控制还要求机器具有可靠性，计算机可以满足这一要求。由于计算机技术的发展，特别是集成电路、大规模集成电路的出现，现代计算机能够连续正常工作几千小时，甚至上万小时。如果用几台计算机并联工作，一台发生了故障，还可由其它几台继续工作。

计算机所具有的上述控制机能使它能有效地应用于自动控制，广泛地应用于各个领域。

本书所介绍的主要是微型机应用于自动控制和使用微型机等方面的实验和制作。动手完成这些应用实验和实用制作，不仅具有一定的实用价值，更重要的是为今后在计算机应用这一广阔天地里施展才能打下基础。

第二章　实验接口技术

本章介绍的是接口技术的基础知识，重点介绍本书实验中常用的APPLE-Ⅱ微型机的游戏I/O接口及其有关控制指令和使用方法。

第一节　微型机接口

和其它控制系统一样，在微型计算机控制系统中，计算机要按照原定要求去监视和控制外部设备（例如一台机床，一个炉子或其它机械设备），发出一系列相应的指令去指挥它的外部设备，使之按一定的要求动作。当系统发生故障或某些参数（如温度、压力等）超过预定指标时，它能及时发出报警信号，或停止外部设备的工作，这就是输出。这种输出可能是动作指令，它告诉外部设备应该如何动作；这种输出也可能是数据，它告诉外部设备动作的具体数量要求，如切削的深度、温度的高低、压力的大小等等。

在多数控制系统中，需要对控制对象和现场不断进行监视和测量。例如在温度控制系统中，首先要将测量到的温度送入计算机，并经软件判断其是否满足要求，然后发出相应的输出控制指令。这种测量和采集数据的过程就是微型机的输入。

在微型计算机系统中，信号的输入和输出常常是通过一种输入/输出接口电路（常简称为I/O接口）进行的。I/O接

口的任务就在于把微型机与外部设备联接起来，使它们能够正确地、协调地进行信息的交换。其基本过程如图2-1所示。

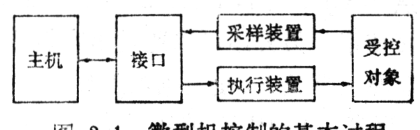


图 2-1 微型机控制的基本过程

在此过程中，当采集装置的性能指标能满足系统的要求时，I/O接口将是决定整个系统速度、精度及控制路数等主要功能的关键。由于各受控对象对系统提出的要求有所不同，常需要自己制作接口电路，在多数情况下，要求接口具有数据转换、地址选择、定时及模拟量和数字量之间的转换功能。接口的这些特点以及采样装置、执行装置等都将在以后的章节中逐步介绍。

第二节 APPLE-Ⅱ微型机简介

由于APPLE-Ⅱ微型机及其国产兼容机具有优良的性能价格比，且有着丰富的硬件和软件，深受国内用户的欢迎，并已被国家有关部门列为中小学的教学用机，所以本书介绍的大部分实验都在APPLE-Ⅱ微型机上完成。本节将简要介绍APPLE-Ⅱ微型机。

一、APPLE-Ⅱ微型机系统概况

APPLE-Ⅱ微型机系统一般由主机、显示器、键盘、磁盘驱动器和打印机组成。它的最大特点是主机内有八个输入/输出插座（又称槽口）。通过插座，用户可以增加或减少

外部设备，灵活方便地选用不同的外部设备。

在APPLE-Ⅱ微型机及其兼容机的存储器中，有固化的监控程序和浮点 BASIC（APPLE SOFT BASIC），开机后用户可以直接使用。另外还可以通过语言板或者通过语言系统软盘经磁盘驱动器输入，使主机进入整数BASIC 语言、APPLE LOGO语言、APPLE FORTRAN 语言或APPLE PASCAL语言等。

二、APPLE-Ⅱ微型机主机板简介

一个微型机系统的最主要部分是主机。主机一般包括微处理器、存储器、输入输出接口等三个部分。打开APPLE-Ⅱ主机的盒盖，即可见到主机板上的各部件（图2-2）。

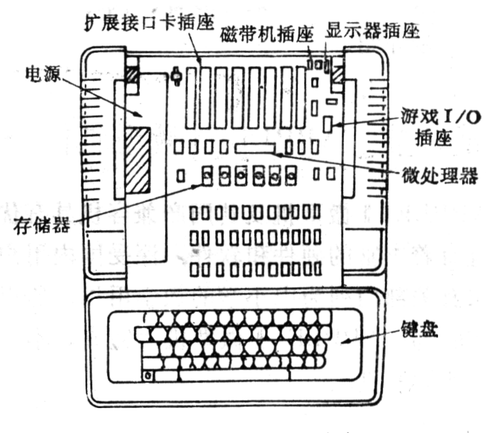


图 2-2 APPLE-Ⅱ主机

1．微处理器：在主机板的后半部分中有一块最大的集成电路芯片，它是APPLE-Ⅱ微型机的心脏——微处理器。微处理器也称微型中央处理器（MPU）。它具有运算和控

制的功能。

2．存储器：存储器是微型机中存放数据、程序和各种信息的部件。它可分成两大类：只读存储器（ROM）和随机存取存储器（RAM）。

只读存储器在主机中位于微处理器的下方，是六块或三块较小的集成电路芯片。它的功能是存储用户不能改动的监控程序和解释程序等，有了这些程序，APPLE-Ⅱ才能正确地理解和执行键盘的命令。只读存储器的内容只允许读出，不允许用户改变，甚至在电源关断后，其中内容也不会消失。

随机存取存储器也称为读／写存储器，它在只读存储器的下方，由三排二十块十六脚的集成电路芯片组成。它的容量一般是48k字节（1k等于1024字节），另外还可以加一块16k字节的语言板，使总容量达到64k字节。随机存取存储器的功能是存储用户经常改变的数据和程序，这些程序就是微型机随时要执行的任务。也就是说，它所存储的内容不仅可读出而且可写入，这些内容当电源关断后会自动消失。

3．电源：主机最左边的金属盒是电源部分。它提供了+5V、-5V、+12V和-12V四种直流电压。它是开关型稳压电源，具有多种保护性能，以保证输出电压的稳定。

4．输入输出接口：输入输出接口是微处理器与外部设备进行信息交换的部件。输入输出接口简称I/O接口。

APPLE-Ⅱ微型机的八个输入输出扩展插座（又称槽口）位于主机板的最上方，每个插座有50个接点。在这些插座上可接入磁盘驱动器、打印机、汉字显示或其它扩充接口板。八个插座自左至右，依次编号为0，1，2，…，7，除了0号插座作为系统本身特别扩展外，其它的插座都可由用户随意使用。用于实际过程控制的一些专用或自制接口

卡，一般都可插在这些插座内。

键盘插座位于键盘下面，在主机板的最下部位。用户通过键盘把所需要输入的数据和程序送进微型机。显示器插座、磁带机插座和游戏控制器插座都位于主机板右侧上方。

这里将作详细介绍的是位于主机电路板右上方的那个不太引人注目的游戏控制器插座，它是一个十六脚双列式插座，旁边的电路板上标有英文字“GAME”（游戏）。在以后的实验中要经常用到这一输入输出接口。

我们知道，微型机控制一般离不开接口电路。采用电路复杂的成品接口或专用接口，固然能得到较多的功能、较高的精度以及较快的采样速度，但是这种接口电路成本高，一般中小学难以添置。另一方面，过于复杂的电子电路也并不适合于向青少年们作介绍。而刚才介绍的游戏接口实际上是一个很有用的现成接口。游戏接口多用于连接游戏操纵杆的微型机游戏。实际上该接口兼有开关量输入、开关量输出、模拟量输入等功能，是适合于微型机自动控制实验的接口。实验使用这样的接口，既不增加投资，又能充分挖掘APPLE-Ⅱ微型机的现有潜在功能。

由于本书以介绍实验内容为主，所以对APPLE-Ⅱ微型机只作有关方面的介绍，要详细了解该机的读者，可查阅有关专著和手册。

第三节　游戏接口及其有关知识

一、开关量和模拟量

日常生活中，人们广泛地使用开关，并应用开关原理。

例如锁、水库的闸门等等就用了开关原理。科学家们研制电子计算机也用到了开关原理。电子计算机，既使是微型机，也要用到成千上万只开关。使用一般的开关显然是不行的，而需要采用电子元件。开始是用电子管、晶体管，现在是用集成电路。这些器件都具有开关特点，即存在着两种状态：高电平或低电平，就象开关的“接通”和“断开”一样。用这样的元件构成的电路叫做开关电路，亦称逻辑电路。

早在十九世纪四十年代，英国数学家、逻辑学家布尔就提供了一个十分简单而又实用的理论工具——布尔代数或称开关代数。在开关代数中只考虑两个量0或者1以及三种运算，其运算定义如下：

1. “或”运算：“或”运算的数学表达式为

$$0+0=0$$

$$0+1=1$$

$$1+1=1$$

“或”运算的符号与普通加号一样，但是意义不同。用一句话概括“或”的意义就是：在“或”运算中，两个变量中只要有一个“1”，其结果就为1。在日常生活中这种“或”的现象很多，例如，楼梯的照明灯，楼上楼下各有一只开关，只要有一个开关闭合（一个变量为1），灯就会亮（结果为1）。

2. “与”运算：“与”运算的数学表达式为

$$0\times 0=0$$

$$0\times 1=0$$

$$1\times 1=1$$

“与”运算的符号与普通乘号一样，但意义是不同的。通俗地讲，在“与”运算中，只有当两个变量同时为1，其

结果才为1。例如，一只保险箱有两把不同的钥匙，只有当两把钥匙同时开锁（全为1时），保险箱才能打开（结果为1），否则就打不开（结果为0）。

3．“非”运算：“非”运算的数学表达式为

$$\overline{0}=1$$

$$\overline{1}=0$$

这种运算是开关代数所特有的，因为在开关代数中只有两个量，不是0即为1，或者不是1就是0。

一个开关有两种状态，分别用0和1来表示，那么一只开关可表示一位二进制。若把n个开关排在一起，则可以表示2^n个不同的数，这样就构成了电子数字计算机中的数字，这种用1和0来表示的数字通常称为开关量。

开关量具有间断的特点，它能够被数字计算机所接受。而我们平时遇到的大部分物理量，如温度、压力、速度等等都是连续变化的量。例如正在加热的室温逐渐上升；压力逐渐变小；傍晚外面逐渐暗下来。这些连续变化的量通常称为模拟量，它不能直接被数字计算机所接受，必须先通过模数转换（A/D）电路将模拟量转化为相应的数字量，再送入数字计算机。

二、软开关

在日常生活中，我们已经接触过不少开关，如电灯开关、电视机开关、遥控电扇开关等。当开启或关闭开关时，就能改变受控电器的状态，但是，在电子计算机出现之前，开关并没有“软”、“硬”之分。

自从电子计算机问世以后，由于深入研究这门学科的需要，把组成计算机的那些看得见、摸得着的装置，如主机、

显示器、打印机等称为硬件。

但是，如果只有硬件，电子计算机并不能工作，就象使用算盘计算时，如果没有运算法则和计算步骤算盘不能正常地发挥作用一样。要计算机顺利地运行并处理各种问题，必须给它编制各种程序。所谓程序，就是为解决某一具体问题而编写的一组指令串。为使用和维护计算机所编制的各种各样的程序，统称为计算机的软件。

在微型机控制中，经常需要开启或关闭多个开关，这些动作都是在程序的指挥下完成的，即这些开关都受软件的控制，因此，我们把这种开关称为“软开关”。相应地，我们把机械或电子开关称为“硬开关”。软开关的开关速度，抗干扰性以及防止开关抖动等都是自动控制领域中经常遇到的问题，需要我们根据不同的受控对象来编制程序，以得到最佳控制。

三、POKE和PEEK

在APPLE-Ⅱ微型机的应用中，怎样用程序来控制软开关呢？控制这些软开关的办法是用6502读写指令对相应的地址进行读写。

软开关的开关状态与内存单元的地址有关 要进入某种状态，就需对相应的内存单元进行访问。在编写一个BASIC程序过程中，可以定义一些变量名，如A、A＄、C1等，以此来处理数据和内存单元之间的联系，但并不了解这些字符所代表的是哪一个内存单元。也就是说，该内存单元的地址是不清楚的，不过一般不需要去了解它，因为BASIC编译程序会自动地安排数据的存放地址。然而，为了充分发挥计算机的效能，有时必须利用某些特定的内存单元，这就需要了

解它们的分配状况及其对应的地址。为了快速而有效地达到这个目的，需使用汇编语言，但也可用BASIC语言中某些命令或语句来把数据读写到指定的内存单元之中，这就是POKE和PEEK指令。其中 POKE 指令是将资料由计算机送至外部设备，PEEK指令是将资料自外部设备送至计算机。这些指令须有一定的格式才能正常工作。

POKE指令必须指明输出口的地址以及数据。该指令的格式为POKE　X，Y，其中X为十进制地址，Y为十进制数据。由于6502微处理器的地址用16位表示，因此可选取65536（2^{16}）个位置，即X值的范围必须在 0 —65535 之 间；而数据用 8 位表示，因此Y值必 须 在 0 —255（2^8）之 间。例 如POKE　12684，215 即 将数值 215 送至地址为12684的输出口。

输入指令PEEK与POKE指令不尽相同。由于该指令欲获得的为输入口的数据，因此PEEK 指令不含数据，只需指明输入设备的十进制地址就够了，它的格式是 PEEK(X)，其中X为输入端的十进制地址。在BASIC语言中，该指令用赋值语句来完成，如A＝PEEK（34579）。由于 6502微处理器为八位机，所以 PEEK 指令所输入的数据应在 0 —255之间。我们经常用PEEK和POKE指令来检查或改变存储器中的数据，所以应该对APPLE-Ⅱ微型机的存储器地址分配加以了解，否则会导致整个微型机工作系统的失败。

四、地址和开关

APPLE-Ⅱ微型机是用 6502 芯片作为它的中央处理单元，它能处理65536个存储单元的工作，对于 64K 内存容量来说，内存单元的地址应在 0 到 65536 之间。但APPLE-Ⅱ

微型机也能认识用实际地址减去65536所得负值表示的地址。例如 49249 这个地址就相当于下面的负值：

$$49249-65536=-16287$$

也就是说PEEK（49249）和PEEK（-16287）是相同的。

POKE指令可以用来控制开关。APPLE-Ⅱ外设接口中的八个软开关就是用POKE指令来控制的。其中四个用于屏幕显示方式的控制，另外四个用于游戏接口的控制。这里先简单讨论一下用于屏幕显示方式的控制，游戏接口的控制将在后一节中作较详细的讨论。

图形 -16304　全屏幕 -16302　第一页 -16300　低分辨率 -16298

文本 -16303　混合方式 -16301　第二页 -16299　高分辨率 -16297

图 2-3　屏幕显示的软开关控制

如图2-3所示，我们用指令POKE -16303，0使屏幕显示方式进入文本显示，因为地址-16303是文本方式的地址，而数据0无关于访问的性质，所以，POKE可把软开关“扳”指向任一个开关状态。在开关“扳”到另一状态位置之前，它会一直停在原处。由一种显示方式转换到另一种显示方式只需改变相应的开关。又如由高分辨率图形第一页全屏幕显示方式转换到第二页，只需改变第三个开关（POKE -16299，0）。再如由高分辨率图形第一页全屏幕显示方式转换到高分辨率图形第一页混合显示方式，就需改变第二个开关（POKE -16301，0）。

五、APPLE-Ⅱ微型机的游戏I/O接口

I/O 接口即输入输出接口。游戏I/O接口位于主机板的右上角，它是一个两边各有八个接脚的长方形插座（见图2-4）。

该接口可以连接特殊的输入和输出元件，如摇杆、按钮、指示灯等。它还常常作为游戏的输入输出插座，游戏程序使

用这一插座后，可以提高使用的效率。图2-4中，AN_0～AN_3端分别为一位输出端，PB_0～PB_2端分别为一位输入端，GC_0～GC_3分别为四个模拟输入端，第1脚为电源电压+5V端，第5脚为数据选通（$\overline{\text{C040 STROBE}}$）接脚，第8脚为接地端，第9、16脚为空脚。

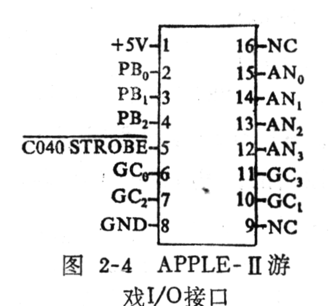


图 2-4 APPLE-Ⅱ游戏I/O接口

1．开关量输出：游戏I/O接口的AN_0～AN_3端分别为一位开关量输出端，每一位都可作为某些电子设备的输入信号，或者作为驱动电路的信号，通过驱动电路推动电子设备。每一位都由一对软开关控制，一个地址对应"关"，另一个地址对应"开"。当开关处于"关"状态时，插座上对应的接脚的电压接近0V；当开关处于

表 2-1

一位输出	状态	地址		
		十进制		十六进制
0	关	49240	−16296	$C058
	开	49241	−16295	$C059
1	关	49242	−16294	$C05A
	开	49243	−16293	$C05B
2	关	49244	−16292	$C05C
	开	49245	−16291	$C05D
3	关	49246	−16290	$C05E
	开	49247	−16289	$C05F

“开”状态时，接脚上的电压约为3.5V。表2-1是四对软开关的地址，图 2-5 是它们的开关状态示意图。

2. 开关量输入：PB_0～PB_2为三个一位开关量输入端，可与电子器件或按钮相连，机器语言和BASIC语言都可以使用。它们的存储单元地址为＄C061到＄C063（十进制49249到49251等效－16287到－16285）。

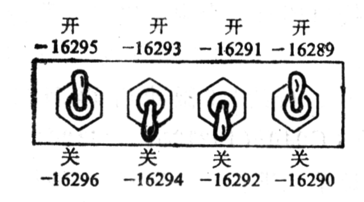


图 2-5 开关量输出的软开关控制

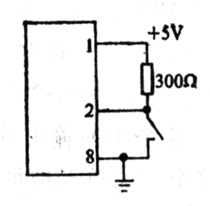


图 2-6 开关量输入

如图2-6所示，将＋5V串接一个300Ω左右的电阻，接到开关量的输入端，同时将此输入端经一开关的常开触点接地。开关状态可由相应的口读入。如果读入数据大于127，则相应的输入端的开关触点断开；如果小于127，则相应的输入端的开关触点闭合。用此开关量输入作为控制对象的状态检测口应用将是十分方便的。

3. 模拟量输入：GC_0～GC_3为游戏I/O接口的四个模拟量输入端，可以和150kΩ的电位器或可变电阻相连。这个电位器的一端接模拟输入端，另一端接＋5V。它实际上接在单稳态电路中，当电阻值改变时，电路的定时时间随着改变。机器语言的程序可以查出这个时间变化，并转换成数值变化。

程序在开始读入电位器变化之前，必须先使单稳态定时电路复位。存储单元＄C070（十进制49264或等效－16272）的软开关即模拟复位开关。复位定时电路复位后，存储单元

＄C064到＄C067（49252到49255或－16284到－16281）四个存储单元的内容都大于128。在3.06ms内，这四个单元的内容值会降到小于128。从复位后的下降时间和该存储单元相连的游戏控制器的电阻变化有关。当模拟输入电位器的电阻值大于150kΩ或没有接电位器时，存储单元内容不会降到128以下。

4．选通输出：游戏I/O接口还有一个附加输出，称为“选通输出”。平时它的输出为＋5V，在机器语言或BASIC程序控制下，选通输出会降到0V，持续0.5微秒后再回升到5V。这个选通输出的地址是＄C040（49216或－16320）。使用这个存储单元地址时，对该单元进行“写”就会有两个选通信号输出。

表2-2列出了游戏I/O接口的开关量输入、模拟量输入、模拟复位以及实用选通的专用存储单元地址。

表 2-2

功　　能	十　进　制		十六进制
开关量输入	49249	－16287	＄C061
	49250	－16286	＄C062
	49251	－16285	＄C063
模拟输入	49252	－16284	＄C064
	49253	－16283	＄C065
	49254	－16282	＄C066
	49255	－16281	＄C067
模拟复位	49264	－16276	＄C070
实用选通	49216	－16320	＄C040

第三章　实验的准备

考虑到部分读者可能对电路不太熟悉，本章的第一节将介绍有关电源和电子器件的基础知识。第二节将介绍完成实验所需的测试仪器和仪表、附加装置以及有关注意事项。因为本书的主要对象是具有一定BASIC语言基础的读者，所以有关语言方面的基础知识，这里不再作详细的介绍。

第一节　电源和电子器件基础知识

一、电源

电源有两种，一种是直流电源，常用缩写DC表示，它的电流方向一定；另一种是交流电源，常用缩写AC表示，它的电流方向随着时间的变化而周期性地变化。直流电幅度与时间的关系图是直线，而交流电幅度与时间的关系图是正弦波，它的频率用Hz（赫兹）表示，我国使用的电源频率为50Hz。

本书介绍的实验较多地用到直流电源，在部分实验中也用到交流电源。前者主要用于控制接口的电子线路，后者主要用于使用220V交流的受控电器。

从220V交流得到直流稳压电源大致需要经过如下过程：变压—整流—滤波—稳压处理—稳定的直流电压；如图3-1所示。图 3-1·a是使用三端集成稳压器的典型电路，图 3-1·b给出了ⓐ点至ⓔ点的电压波形。

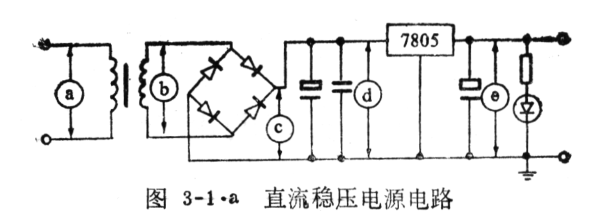


图 3-1·a 直流稳压电源电路

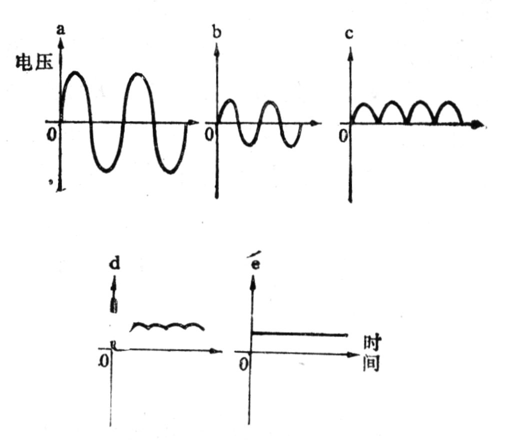


图 3-1·b 直流稳压电源电路中的波形

二、电子器材

下面介绍常用的电子器材。

1．二极管：二极管有正极和负极两个极，它具有单向导电性，电流从其正极流向负极。

二极管分整流、检波、开关、稳压、光敏等多种类型，它们的功能各不相同，被应用到多种电路中。

2．三极管：三极管的三个极分别是基极、发射极和集电极。如果按制造材料分，三极管可分为硅管和锗管，如果按极性分可分为PNP型和NPN型，如图3-3所示。

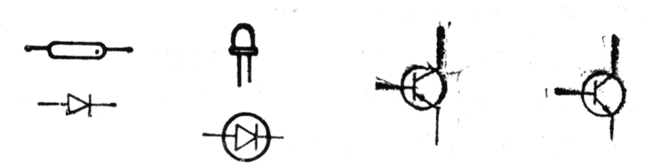


图 3-2 二极管　　　　　　　　图 3-3 三极管

图3-3中，三极管电路图内箭头向里的为PNP型，箭头向外的为NPN型。

当三极管的b、e之间有基极电流流过时，c、e之间就会有大得多的电流流过，这种电流即为集电极电流，它等于基极电流的h_{fe}倍，这相当于把基极电流放大了h_{fe}倍，h_{fe}即为三极管的电流放大系数。但是集电极电流并不能无限增大，每种型号的晶体三极管都有它的最大集电极电流的规定。

三极管按其功率大小，可分为小功率管、中功率管和大功率管，它们的外形如图3-4所示。

小功率晶体管的集电极电流较小，不能直接驱动继电器，需要在后面加一级功率驱动电路，来驱动继电器等负载。

3．电阻：电阻是一种对电流起阻碍作用的元件，它的基本单位是欧姆（Ω），经常使用的单位还有千欧（kΩ）等。根据不同的需要，还有功率和精度之分。

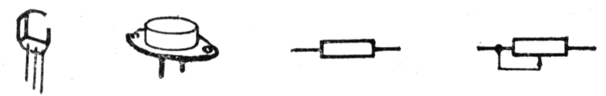


图 3-4 三极管的外形　　　　　　图 3-5 电阻

4．电容器：电容器的种类较多，根据制作的材料不同可分为电解电容、瓷片电容、金属膜电容等等。根据用途可分为滤波、旁路、谐振电容等等。电解电容器有正极和负极之分，其它电容器没有极性。电容的基本单位为法拉，常用的

单位为微法（μF）和微微法（pF）。

5．开关：开关的种类繁多，这里使用较多的有按钮开关，单刀开关和拨动开关等。对接至220V交流电的开关，应注意选择合适的耐压和触点容量。这些开关的电路符号如图3-7所示。

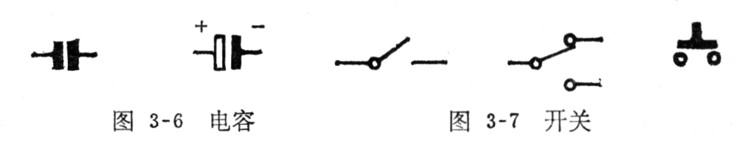


图 3-6　电容　　　　图 3-7　开关

6．继电器：继电器及其电路符号如图3-8所示。

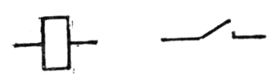


图 3-8　继电器

继电器有交流和直流之分，选用时要加以注意。

用电子线路控制电器时，常在晶体管或集成电路的功率驱动电路中接入继电器，如图 3-9 所示。

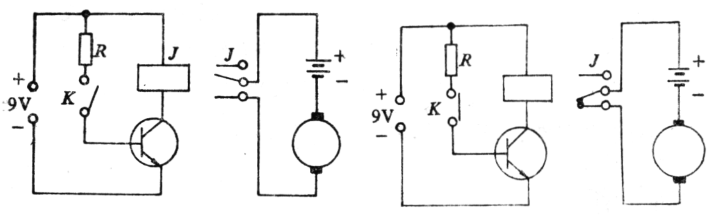


图 3-9a　控制马达停　　　　图 3-9b　控制马达开

图 3-9 是一个控制马达开停的基本电路图。继电器 J 的线圈接在三极管的集电极回路中，马达的工作电压回路中串接进继电器的常开触点。当开关K断开时，三极管的基极没有电流流过，集电极也无电流，此时继电器释放，马达停止转动，如图 3-9a 所示；当开关 K 闭合时，三极管得到了足

够的基极电流而导通，其集电极电流使继电器吸合，马达旋转，如图 3-9b 所示。

继电器的种类很多，即使是同一种型号的继电器，可供选择的工作电压也有多种，如48 V、24 V、12 V、6 V等等。各类继电器的触点负载能力也常有不同，例如常用的 JRX-13F-1型小型小功率继电器的触点负载为48V交流，0.5 A，而JQX- 4 型继电器的触点负载为220V交流，3A。为了使实验电路更简化一些，我们较多地使用JQX-4型继电器。

7. 集成电路：将许多晶体二极管、三极管及电阻、电容制作在同一芯片上的器件称作集成电路，常简称为IC。按其集成度可分为小规模集成电路，中规模集成电路以及大规模和超大规模集成电路。按其用途可分为数字集成电路，模拟集成电路及专用集成电路。目前，集成电路的应用极其广泛，几乎渗透到各个领域。

尽管集成电路的制造工艺以及内部结构要比晶体管复杂得多，但是使用起来却十分方便。只要我们的总体电路设计合理，并对所用集成电路的技术性能、使用要求、封装方式以及它的接脚功能等了解清楚，使用起来就会得心应手。

集成电路有多种封装形式。常见的有双列直插封装、单列直插封装、扁平封装、金属圆壳封装、三端封装等，其中较早采用的双列直插封装，由于安装灵活，使用方便，仍然是应用最多的一种。如图3-10所示。

对于双列直插封装应注意其正面的缺口所在。把缺口放到左边，然后从它的左下角的第一个脚开始，逆时针方向从 1 数起即为它的接脚号码。也有部分集成电路的正面用明显的色点标出第一脚的位置。总之，必须先对照产品说明，把接脚搞清楚以后才能正确使用。图3-11是集成电路引脚排列示

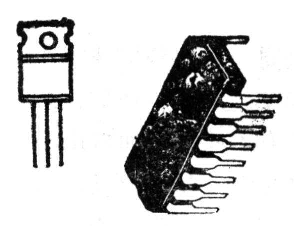


图 3-10 集成电路

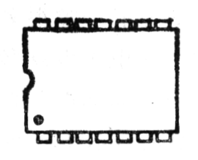


图 3-11 集成电路的引脚

本书介绍的微型机实验中所使用的集成电路主要有反相器、功率驱动器、集成稳压器、译码器、运算放大器、温度传感器以及555时基电路等。这些集成电路将在有关的章节中逐一介绍。

第二节 实验所需的设备

一、微型机系统

一般的微型机基本系统，除了配有主机、键盘和显示器外，还应该配有驱动器和打印机，其中，驱动器也可以用录音机来代替。

本书介绍的大部分实验都在APPLE-Ⅱ微型机的游戏I/O接口上完成。由于次数较多，使用该接口时，应该在原来焊牢在电路上的十六脚双列插座上再插入一个保护插座，这样即使因为插拔次数较多而引起插座簧片松开时，也可以方便地调换，而毫不影响主机电路板上的原插座。

使用其它接口板的实验，无论购置或自制接口，都应保证接口板的质量以及与主机连接的正确性，以免损坏主机或给实验带来不必要的麻烦。

二、电路实验板

这里介绍的多数实验需要制作有关的控制接口电路。我们选用的电路实验板是专门为电子线路实验而设计的，它的基板主体为白色绝缘材料，上面整齐地排列着一排排方孔，每个方孔内都嵌有富有弹性的铜质簧片作插脚座，使插入孔内的电子元件和连接线接触良好。电路板的中部还有一条宽度适中的横槽，可使双列直插型集成电路正好跨插在它的两侧。同时该横槽又把整个实验板分成上下两个部分，每一部分的纵向是一铜片槽，将同一列上的各点互相连接在一起，这样可使连接线大为减少。电路实验板的示意如图3-12所示。

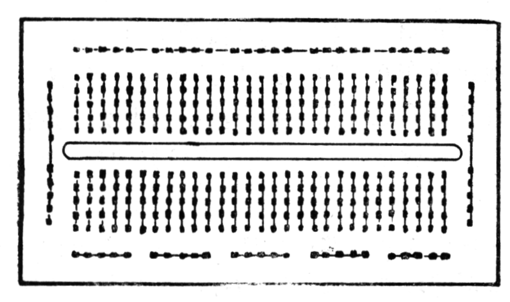


图 3-12 电路实验板

同制作印刷电路板相比较，虽然实验板的接触可靠性比较差，电路的连线也比较零乱，但是，使用实验板时，修改电路、更换元件较方便，这样可以节省很多时间，因而这种实验板在实验室中仍然得到了广泛应用。

三、万用表

在控制接口电路的制作以及微型机的输出电平监测中都需要使用检测仪表。万用表是一种价格不高、用途广泛的简

易仪表。它可以用来测量电压、电流、电阻等等。多数万用表还可以用来测量三极管的电流放大倍数及泄漏电流。价格较高的数字式万用表的主要性能优于指针式万用表，并且能对电容、温度等进行测量。

四、直流稳压电源

直流稳压电源是电子设备的能源部件，其性能良好与否直接影响整个设备的精度、稳定性和可靠性。我们知道，大多数电子器件如晶体管、集成电路等都在低压直流条件下工作，可以这样认为，直流稳压电源是电子线路实验所必不可少的基础仪器。

本书所介绍的实验要求直流稳压电源具有如下性能：它的输出电压应在 0～24V之间连续可调，负载电流应达2A以上，面板分别有电压指示和电流指示表头，并有可靠的过流保护措施和过载指示，电源还应有较好的稳定度和低的纹波。一般市售的中档直流稳压电源都能满足以上要求。随着三端可调集成稳压器的普及，一些业余爱好者设计的可调直流稳压电源也都能达到上述性能指标，具体制作方法详见第六章第一节。

五、示波器

示波器是一种用途极其广泛的电子测量仪器，它能够比较清晰地显示出微微秒数量级的脉冲波或者变化极其缓慢的非周期性的电压变化，以便精确地进行定性、定量的分析和研究。示波器之所以应用广泛，是由于它能够把原来非常抽象的、肉眼看不见的电过程，变换成具体的看得见的图象。

在以后的实验过程中，我们将用示波器测量脉冲波形、

频率、电压、电流、相位等。它既可以查找电路中可能出现的故障，也可以帮助我们更好地学习和理解电路。如果配上一些外接电路，示波器还能对晶体管、可控硅以及某些集成电路进行测量。

以上介绍的实验仪器都是最常用的电子测量仪器。由于实验所用的电子器件形形色色，因此，在有条件的学校，如果还备有晶体管特性图示仪、集成电路测试仪、阻容测量仪等，将会给实验带来更多的方便。

需要说明的是，本书所介绍的实验并非都要使用上述仪器，不少实验只需几件简单的工具就可动手完成，有的实验虽然要求使用仪器，但是如果实在没有这些仪器，也可完成实验。

第四章　微型机应用实验及有关装置

本章通过二十多个微型机应用实验的例子，着重介绍微型机的接口技术。通过这些实验，读者不仅可以学到许多有关接口的硬件知识以及制作方法，而且可以进一步巩固和加深所学的软件知识。

第一节　用“软开关”开关电器

在第二章我们已经学过APPLE-Ⅱ微型机游戏 I/O 接口的开关量输出控制指令，运用这些知识，再编写若干简单的BASIC程序，就可以进行控制家用电器的实验了。

一、软开关状态的仪表检测

大家知道，APPLE-Ⅱ游戏 I/O 接口可用于游戏程序中的附加控制装置，但是，对于该接口的其它功能和应用，不少人并不很熟悉，通过控制家用电器的实验，可使我们对该接口有一个更全面的认识。

做实验之前，先让我们用仪表对接口的四个开关量输出进行检测。为了避免损坏主机电路板上的16脚双列式插座，我们可先在该插座上再插入一个同样的插座，因为在实验中将多次使用游戏 I/O 接口，容易使插座的簧片失去弹性。

先判别一下插座上的序号，注意插座上的缺口方向。然

后将一根细的单股导线，一头插入该插座的第8脚（接地端），把它的另一头接至万用表的负端（黑表笔）。用另一根导线的一头插入插座的第15脚（标有AN_0处），把它的另一头接至万用表的正端（红表笔），并将万用表的功能选择开关旋至直流（DC）10V档上，然后开启主机，并从键盘上输入程序4-1：

程序4-1

```
10 POKE -16295,1
20 GOTO 10
```

对照软开关的地址表可以看出，程序4-1的10语句是软开关的开语句，它使AN_0端输出高电平。在执行RUN指令以后，万用表上的指针指向+5V。然后将程序修改为：

程序4-2

```
10 POKE -16296,0
20 GOTO 10
```

对照软开关的地址表可以看出，修改后的程序4-2中10语句是软开关的关语句，它使AN_0端输出低电平。在执行RUN指令以后，万用表上的指针回到0V。输出的高电平+5V和低电平0V分别表示了开和关两个状态。

为了进一步验证地址和开关之间的对应关系，我们改换一个输出端，把原来插入第15脚的一端插入第14脚，并改写程序为：

程序4-3

```
10 POKE -16293,1
20 GOTO 10
```

RUN以后在万用表上指示出高电平，然后修改程序使之为：

程序4-4

```
10 POKE -16294,0
20 GOTO 10
```

RUN以后在万用表上指示出低电平。由此可见，当程序中的地址改变时，软开关所控制的输出端也相应地改变了。

我们也可以用示波器来检测游戏I/O接口的开关量输出端。将示波器的输入探头接至接口的开关量输出端，示波器的功能选择开关旋至DC档上，然后运行以上程序，也可以清楚地看到它的输出电平的变化。

二、接口驱动电路简介

怎样用计算机的软开关技术来控制家用电器？

前面已经讨论过，APPLE-Ⅱ游戏I/O接口并不能直接控制电器，必须通过它的接口输出电路来控制。图4-1是它的输出接口电路图。

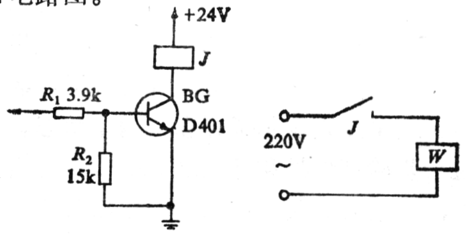


图 4-1 三极管开关电路

这是一个基本的三极管开关电路。当它的输入端接高电平时，三极管迅速导通，串接在它的集电极电路中的继电器线圈有电流通过，继电器J吸合，它的常开触点接通，220V交流电压加到电器W上。当电路的输入端接低电平时，三极

管迅速截止，继电器线圈中无电流通过，它的常开触点又断开，电器W又失电而停止工作。

这里的晶体管BG选用D401大功率塑封管，也可以选用另一些功率晶体管。继电器J选用JQX-4型，由于它的触点负载为220V交流、3A，所以可直接控制一般的家用电器。当然也可以选用其它型号的继电器，但必须注意它的工作电压及触点的负载能力。

第二节　教学软件中的有声系统

一、计算机辅助教学简介

计算机辅助教学简称CAI，是从六十年代开始发展起来的一种新的教育技术，世界上一些发达国家把它作为改革教育的一项重大措施。早期的CAI以大、中或小型计算机为基础，因其成本较高而不便于推广。从七十年代中后期开始，微型计算机逐渐进入CAI领域，现在发展以微型机为基础的CAI已成为主流。

有的计算机辅助教学软件只有图象和文字而无发声系统。有的既有图象和文字，又有声音讲解，显然后者的效果要好些。目前国内已有科研单位研制成功采用配音接口卡的有声系统。但是配音接口卡的电路较复杂，系统成本高，因此，这里介绍一种方法，利用APPLE-Ⅱ微型机的游戏接口作为控制接口，加入有声系统。尽管它的功能没有采用配音接口卡的有声系统功能强，但其成本很低，制作容易，仍然不失为能在中小学推广的一种方法。

二、有声系统简介

APPLE-Ⅱ微型机也有发声功能，早期也有人用6502汇编语言来编写发声程序，但是发出的汉语失真大、音量小，学生不易听清楚。而且这种发声程序必须编得很长，相当耗费时间，修改也很不方便。

前面已经介绍了游戏I/O接口的四个开关量输出端，以及控制其输出电平的软开关语句，再自制简单的输出控制电路，就可以方便地开启或关闭录音机了。这样我们可以事先把教师的讲解录制在磁带上，并在原来的无声程序中需要配音的地方加入开启录音机的软开关语句，而在不需要发声的时候加入关闭录音机的软开关语句，这样就得到了简易有声系统。应该注意到，发声系统必须同程序中加入的软开关语句同步。只要我们细心录制，这一点是完全有保证的。在一些要求较高的发声系统中，可以配用两台或更多的录音机。

三、控制电路及其制作

简易有声系统的输出控制电路是由两个干簧继电器及一些插头、插座组成，其电路如图4-2所示：

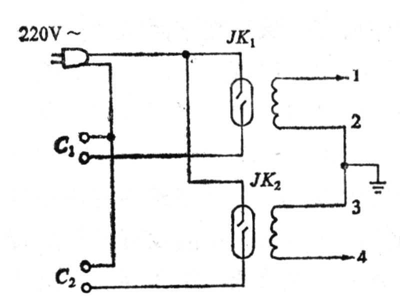


图4-2　有声系统的输出控制电路

图4-2中JK_1、JK_2是两个干簧继电器，它们可以用干簧管自制，具体方法是：根据所使用的录音机的工作电流选择干簧管，再用0.08—0.1mm的漆包线在干簧管外面绕500圈左右。为了防止漆包线从管

壁上脱落，最好事先制作一个绝缘骨架，套在干簧管两头，并将干簧管与骨架装紧，然后再绕漆包线。绕好的干簧继电器可按图 4-2 与电源插头及两个电源插座C_1、C_2相连组成一个外电路控制硬件。装配时一定要注意，漆包线同连接线的绝缘要可靠。

使用时，只需要将图4-2中标有 1 和 4 的两端引线插在AN_0～AN_3中的任意两个脚上（如干簧管不配套，可在 1 、2 两端上各加一前置继电器后再接在游戏接口上），并将 2 和 3 端连接后，接在共同的地端即游戏 I/O 接口的第 8 脚上；将录音机的电源插头插在C_1、C_2插座上，并按下放音键。在程序未运行之前，由于 1 端同 2 端之间没有电流流过，JK_1的簧片未吸合，插在C_1插座上的录音机由于一端断开停止工作。程序运行后，当软开关的开语句出现时，相应的开关量输出端出现高电平，由于 2 端接地， 1 端和 2 端中的线圈有电流流过，干簧管吸合，录音机电源接通，开始放音。

这套控制设备的制作并不难，加入微型机系统后也不会影响APPLE-Ⅱ微型机的性能，更不会影响你用 APPLE-Ⅱ微型机来完成其它课题。如你原来使用的程序中没有有声系统，只要在程序中加入简单的软开关语句后就可以使用以上介绍的有声系统了，无需对原有的程序作其它改动。

四、计算机有声阅卷

下面的例子是让计算机出题并阅卷，然后将阅卷结果通过有声系统告诉你。这里使用两台录音机，其中一台录有“你做对了”的录音，将它的电源插头插入C_1插座，同时将图4-2中标有 1 的这一端插入游戏I/O的第15脚；另一台录有“你做错了，再做一遍”的录音，将它的电源插头插入C_2插

座，同时将 4 端插入游戏I/O的第14脚。下面介绍两个程序，其中程序4-5用于无声系统，而修改为程序4-6后就可用于有声系统了。试加以比较并完成这个实验。

程序4-5

```
10  A=  INT(100*RND(1))
20  B=  INT(100*RND(1))
30  PRINT A;"+";B;"="
40  INPUT C
50  IF A+B=C THEN 80
60  PRINT"WRONG!DO IT AGAIN!"
70  GOTO 10
80  PRINT "RIGHT! NEXT ONE PLEASE!"
90  GOTO 10
100 END
```

程序4-6

```
10  A=  INT(100*RND(1))
20  B=  INT(100*RND(1))
30  PRINT A;"+";B;"="
40  INPUT C
50  IF A+B=C THEN 70
52  POKE  -  16295,1
54  FOR I  =  1 TO 4000:NEXT I
56  POKE  -  16296,0
60  GOTO 10
70  POKE  -  16293,1
72  FOR I  =  1 TO 4000: NEXT I
74  POKE  -  16294,0
80  GOTO 10
90  END
```

第三节　软件定时器

一、定时器

我们几乎每天都要和定时器打交道。一般的家用洗衣机都装有定时器，调节定时器的旋钮，可使洗衣机在规定的时刻自动关掉，闹钟里也有定时装置，它可按照人们的要求准时鸣响。这些定时装置大多采用了机械定时的方法，称为机械定时器。还有一种采用晶体管或集成电路所组成的时间继电器也能起到定时作用，它们利用了电子器件的时延特性，称为电子定时。电子定时器比机械定时器的使用更为方便，目前在工业控制中已经得到了广泛的应用。这两种定时器都属硬件定时的范畴。

随着微型机应用的日趋广泛，采用程序设计语言的软件定时技术已在过程控制中得到了更多的应用。运用机械或电子定时的方法，需要人工调节预置时间的旋钮，而运用软件定时技术，如需要改变受控时间，只需改变程序中的语句或DATA语句中的数据。数据选择越大，其受控时间越长，所以能轻而易举地得到机械或电子定时器所不能完成的长时间定时。它的定时精度也比硬件定时高，而且还兼有记忆、存储及打印等功能。

二、软件定时

我们经常会碰到这样的情况，某些程序在运行后经过几秒、几分甚至更长的时间后其运算结果才从显示器上显示出来。这是因为每一个程序的运行都需要消耗一定的时间。

微处理机执行指令时，把一条指令分成若干个微操作，顺序完成这些微操作的综合，就是一条指令的动作。微操作由微命令控制，微命令通常是一些脉冲信号。定时信号由外部振荡器产生，然后引入微处理机，它是微处理机一切操作的基本时钟和基本信号。因此，数据的存取、传输和运算都会消耗时间，在科学计算、数据处理时，总希望这种时间越短越好，而在过程控制应用时，有时为了使采集的数据更准确或消除开关抖动等，往往在程序中特意加入延时程序。

先让我们在显示器上演示一下软件的延时功能。从键盘上输入程序4-7。我们可以看到在程序运行后约一秒钟，其结果A=9便显示出来了。

程序4-7

```
10  LET B=3
20  LET A=  B^  2
30  FOR I=1 TO 750:NEXT I
40  PRINT  "A=";A
50  END
```

如果将语句中变量I的循环次数750改为7500后再运行一次，将会看到约10秒以后其结果A=9才显示出来。程序中的30语句是对变量I进行空循环语句，用它可以得到一定时间的延时。增加循环次数，便可得到与次数成正比的延时。

三、电器的定时控制

先让我们用万用表来监测一下软件延时功能。将万用表的红表笔接至游戏I/O插座的第15脚，黑表笔接至第8脚，然后输入程序4-8：

程序4-8

```
10  POKE  -16295,1
20  FOR I  =1 TO 3000:NEXT I
30  POKE  -16296,0
```

在RUN以后可以看到万用表的表针先指示在5V上，经过约4秒钟后又回到0V。再修改程序，使20语句中的循环次数为7500，RUN以后可以看到表针经过约10秒后才回到0V。

利用上述原理可设计一个通用的定时控制程序。因为得到1秒时间所需的空循环次数约为750，所以，只要将所需定时的时间（以秒为单位）乘以750就可以得到总循环次数。现在让我们讨论一下程序4-9。

程序4-9

```
10  INPUT H,M,S
20  LET N=(H*60*60+M*60+S)*750
30  POKE  -16295,1
40  FOR I  =1 TO N: NEXT I
50  POKE  -16296,0
60  END
```

程序中的30语句和50语句分别为控制AN0开和关的软开关语句。

将图4-3所示的输出控制电路接至游戏I/O接口将电器

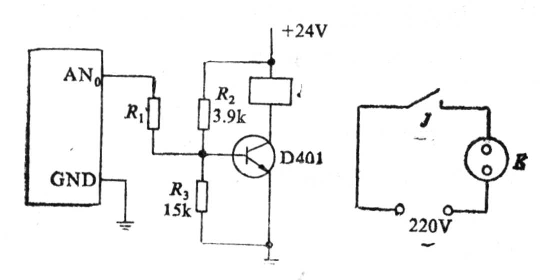


图 4-3　定时控制的输出接口电路

的开关开启后再插入电源插座K，然后输入并运行程序，所需的定时时间从键盘上输入，可以看到按回车键后电器即开始工作，待预定时间一到，电器即失电而停止工作。图4-3中的R_1值需经试验后确定。

第四节 时间控制

本节将介绍用APPLE-Ⅱ微型机进行时间控制的实验。与传统的电子或机械式时间控制装置相比，微型机控制时间的精度高、可靠性好、改变受控时间不需要改变硬件的状态，只需改变程序中的DATA数据，且其输出控制接口电路简单，自制方便，如果将学校的电铃总开关装入机房，便可由微型机来控制作息时间了。

一、APPLE-Ⅱ时钟

前面我们已经学习过软件延时的原理，在APPLE-Ⅱ微型机上，用语句FOR I=1 TO 750:NEXT I可得到约1秒的延时。如果我们用条件或循环语句反复调用这条语句，然后把每次得到的时延累加起来，并且按照时钟的要求，逢60秒化为1分，60分化为1小时，24小时后又从零开始，并在程序中加入显示输出语句，便可以在显示屏上得到一个数字显示的电子钟。程序4-10即为一数字时钟程序：

程序4-10

```
10  REM APPLE CLOCK
20  PRINT  "H=";:INPUT H
30  PRINT  "M=";:INPUT M
40  PRINT  " S=";:INPUT S
```

```
50  FOR K=1 TO 750:NEXT K
60  S=S+1
210 IF S=60 THEN 250
220 IF M=60 THEN 260
230 IF H=24 THEN 270
240 GOTO 280
250 S=0:M=M+1:GOTO 220
260 M=0:H=H+1:GOTO 230
270 H=0
280 HOME:PRINT:PRINT:PRINT:PRINT:PRINT
285 PRINT H;":";M;":";S
290 GOTO 50
```

其中H表示时、M表示分、S表示秒，20～40语句请你从键盘上输入时钟的起始时间，50语句得到1秒的延时，并在60语句中得到累加，210语句～285语句将累加的秒数转换成分和时，并送至显示屏显示出来，这个数字显示的电子时钟就是这样得到的。这个时钟还可以调节快慢。如果将50语句中的循环次数增加，时钟就变慢，反之时钟则变快。

二、时间程序控制

在APPLE-Ⅱ时钟的基础上，我们在程序中加入条件比较语句和软开关语句，可设计出能对多个预置时间进行程序控制的软件。

表4-1是某学校的作息时间表，其中7:27:00表示7点27分0秒。原先由传达室工作人员根据该表开关电铃，大家都知道，这样不仅麻烦，而且有时难免出错。现在有了电子计算机，我们就可以把这个任务交给它去完成。

程序4-11是根据该作息时间表编制的电铃程序控制软

表 4-1

铃声开始	铃响结束	说　　明
7:27:00	7:27:20	预备铃
7:30:00	7:30:10	第一节课开始
8:15:00	8:15:10	第一节课结束
8:25:00	8:25:10	第二节课开始
9:10:00	9:10:10	第二节课结束
9:25:00	9:25:10	第三节课开始
10:10:00	10:10:10	第三节课结束
10:20:00	10:20:10	第四节课开始
11:05:00	11:05:10	第四节课结束
13:17:00	13:17:20	下午第一节课预备
13:20:00	13:20:10	第一节课开始
14:05:00	14:05:10	第一节课结束
14:15:00	14:15:10	第二节课开始
15:00:00	15:00:15	第二节课结束
15:10:00	15:10:10	第三节课开始
15:55:00	15:55:10	第三节课结束

件。

程序4-11

```
10  REM TIMER AND CONTROLLER PROGRAMMING
15  DIM Q$(5)
20  PRINT "H=":INPUT H
30  PRINT "M=":INPUT M
40  PRINT "S=":INPUT S
45  PRINT "Q$=":INPUT Q$
50  IF Q$="START" THEN 60
55  PRINT "INSTRUCTION ERROR":GOTO 45
60  I=I+1
70  READ H1,M1,S1
```

```
75  GOSUB 200
80  IF H=H1 AND M=M1 AND S=S1 THEN 100
90  GOTO 75
100 IF I/2=INT(I/2)THEN 125
115 POKE  -16293,1
120 GOTO 60
125 POKE  -16294,0
130 GOTO 60
140 DATA 7,27,0,7,27,20,7,30,00,7,30,10,8,15,0,8, 15,
    10
141 DATA 8,25,0,8,25,10,9,10,0,9,10,10,9,25,0, 9, 25,
    10
142 DATA 10,10,0,10,10,10,10,20,0,10,20,10, 11, 5, 0,
    11,5,10
143 DATA 13,17,0,13;17,20,13,20,0,13,20,10, 14, 5, 0,
    14,5,10
144 DATA 14,15,0,14,15,10,15,0,0,15,0,10,15,10,0, 15,
    10,10,15,55,0,15,55,10
150 END
200 FOR K=1 TO 750:NEXT:S=S+1
210 IF S=60 THEN 250
220 IF M=60 THEN 260
230 IF H=24 THEN 270
240 GOTO 280
250 S=0:M=M+1:GOTO 220
260 M=0:H=H+1:GOTO 230
270 H=0
280 HOME:PRINT:PRINT:PRINT:PRINT:PRINT
285 PRINT H;":";M;":";S
290 RETURN
```

为了使程序更简洁，我们把每一次的电铃开启和关断时间放在DATA语句中的相邻位置，铃响开始的时间排在奇数位置，铃响结束的时间排在偶数位置，通过100语句中的比较条件将它们分出，然后再分别由115语句或125语句中的软开关语句去控制输出接口电路。

程序输入并运行后，由操作者从键盘上输入起始时间，然后再键入起动指令“START”，此时，程序即进入电子时钟显示的自动程序控制状态。由于这个程序中以秒为最小时间单位，每隔1秒钟，实际时间将同DATA语句中的预置时间进行比较，如果不相同则继续计时，如果相同则发出相应的控制指令，即开启或关断电铃，然后继续进入计时和比较，这个过程一直进行到DATA语句中的预置时间读完为止。

三、输出接口电路

在实验中所用的电铃控制的输出接口电路与上一节图4-3所示的电路完全一样，这里就不再赘述了。

第五节　电铃控制装置

上一节介绍了根据软件延时原理制作的APPLE-Ⅱ时钟。本节将介绍另一种应用微型机控制时间的实验。下面以LASER-200微型机为例，说明如何利用这一类机型的特有功能，制作电铃控制装置。通过本节的实验，可以获得一些新的知识和启迪。

LASER-200微型机电铃控制装置由主机、显示器、录音机及电子控制接口组成，其控制软件用BASIC语言编写。

该装置的最大特点是利用了LASER机的SOUND发声语句，由LASER-200 机 发出音频信号，经电子线路放大处理后触发电子开关，同上一节另有不同的是，这里的铃响开始时间由程序控制，而铃响的延时时间由电子线路完成，因此铃响的结束是由外加电路控制的。

一、程序简介

图4-4是该电铃控制装置的程序框图。下页是程序清单：

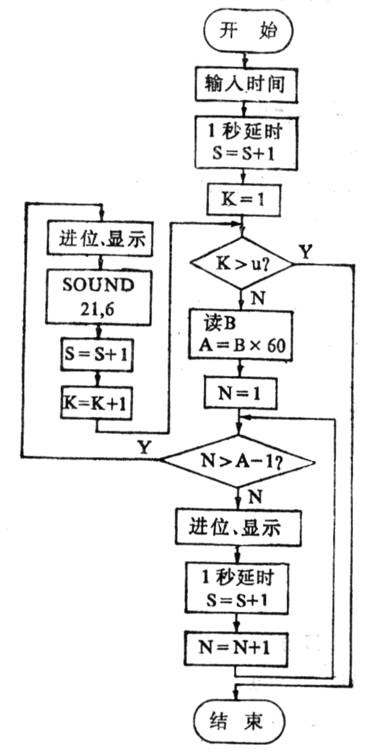


图 4-4 电铃控制的程序框图

程序4-12

```
     LIST
5    REM TIMER AND CONTROLLER PROGRAMMING
10   PRINT "H=":INPUT H
15   PRINT "M=":INPUT M
20   PRINT "S=":INPUT S
30   FOR I=1 TO 450:NEXT
35   S=S+1
40   FOR K=1  TO U
50   READ B
60   A=B*60
70   FOR N=1 TO A-1
80   GOSUB 210
90   FOR I=1 TO 450:NEXT
100    S=S+1
110  NEXT N
120  GOSUB 210
130    SOUND21,6
140    S=S+1
150  NEXT K
160  DATA……
170  DATA……
180  DATA……
190  DATA……
200  END
210  IF S=60 THEN 250
220  IF M=60 THEN 260
230  IF H=24 THEN 270
240  GOTO 280
```

```
250 S=0:M=M+1:GOTO 220
260 M=0:H=H+1:GOTO 230
270 H=0
280 CLS:PRINT:PRINT:PRINT:PRINT:PRINT
285 PRINT"      ";H;":";M;":";S
290 RETURN
```

在主程序中，用循环语句中适当的循环次数得到所需的延时。这里用语句FOR I=1 TO 450:NEXT得到1秒延时时基。应该注意的是，每台LASER-200机得到1秒时基的循环次数稍有不同，应通过程序运行中的调试来选取最佳值。这里所需延时的时间以分为单位置入DATA数据中，由语句A=B*60换算成秒后再反复调用1秒时基，达到预定时间后即退出循环，并执行SOUND 21，6语句，向电子控制部分发出持续时间为1秒的音频信号，此延续时间已由140语句考虑并累计。为了便于监视主机的工作状态，使得整个控制过程更加直观，将不断循环的秒次数累计并转换成分、时显示在荧光屏上。子程序中的210～290语句就是完成这一功能的，并用条件语句将60秒换算为分、60分换算为时，因而在荧光屏上出现的是每秒显示一次的数字钟。

二、电子控制部分

自制的电子控制部分，实际上是接受音频信号控制且具有一定延时功能的电子开关。它的输入端接在主机内部的压电陶瓷片的音频输出端上。图4-5·a是它的电原理图，图4-5·b是它的直流12V稳压电源部分。

电路的工作是这样的：在主机无输出信号时，BG10截止，此时J_{2-1}接点处断开状态，BG4处导通状态，其集电极

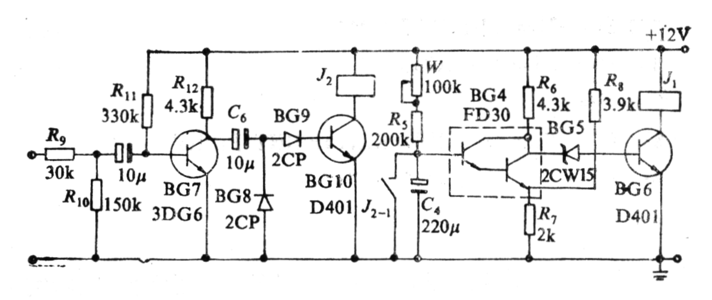


图 4-5·a 电子控制部分的电路

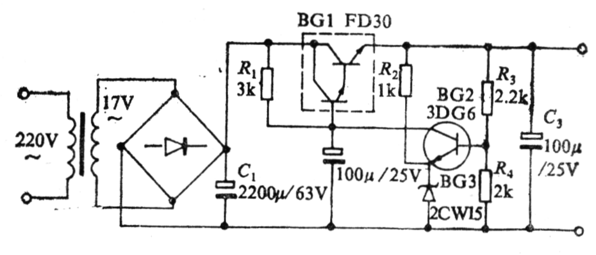


图 4-5·b 直流稳压电源

电位很低，BG5的稳压值在7V左右，处截止状态，所以BG6也截止。

当主机输出音频控制信号时，经BG7一级放大并检波后使BG10导通，J_2吸合，J_{2-1}接点接通，使BG4迅速截止，于是BG5导通，BG6也导通，J_1吸合，电铃开关接通。

1秒音频信号结束后，J_{2-1}又断开，但是，由于W、R_5及C_4组成的延时电路，BG_4的基极电位逐渐上升，经过一段时间延时后BG4导通，BG6截止，J_1释放，铃响结束。其延长时间由W、R_5及C_4的值所确定。图4-5·a所示的数值，其延时时间约在10～30秒之间连续可调。欲得到校长的延时，可增加C_4容量或R_5的电阻值。应注意选择小讯号放大特性好的达林顿管及漏电小的电解电容器。

为使控制装置的工作更为可靠，这里选用了达林顿晶体管作调整管的稳压电源电路，其特点是具有较高的稳定度，同时它的温度性能好，保证了输出接口装置能在220V电网电压变化+10%及-20%的情况下均能正常工作。

三、控制装置与外电路的连接

图4-6为该装置与手动控制电铃的外电路的连接图。安装前应首先切断总电源，然后方可连接电路。将接至手动开关K_1的一端（接电铃端）断开，接至J_1的常开触点，再将同一组的另一个常开触点再接至K_1上的断开端，仔细检查连接无误后可合上总电源，并将K_1放在接通位置上，此时，电铃的控制就由微型机来完成了。

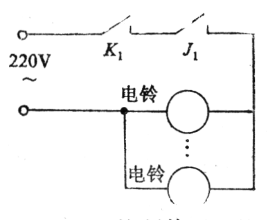


图 4-6　控制装置与外电路的连接

第六节　微型机频率时钟的遥控及应用

利用微型机进行时间控制的途径很多。本节以PC-1500微型机为例，介绍如何利用这类微型机所具有的时钟频率和遥控性能，来制作时控装置。

PC-1500微型机内部的芯片具有时钟频率的特点，一经振荡后，无论是开机状态还是关机状态都能维持时钟频率的振荡。而PRINT #变量语句能遥控外设动作，如果我们在需要动作外设的时间来执行PRINT #变量语句，则达到了对时间进行遥控的目的。也就是在动作时钟频率的某一个脉

冲来到时，PC-1500机的遥控门打开，外部设备动作。在动作时钟频率脉冲过了之后，机内的遥控门关闭。外部设备失压停止动作。当然也可以通过一定的语句方法来延长或缩短遥控门开启的时间，以满足实际工作的需要。这也相当于晶体二极管的开关特性，正向电平时导通，反向电平时截止。只是此处的正、反向电平是由计算机的遥控命令来代替。电路示意图如图4-7所示。

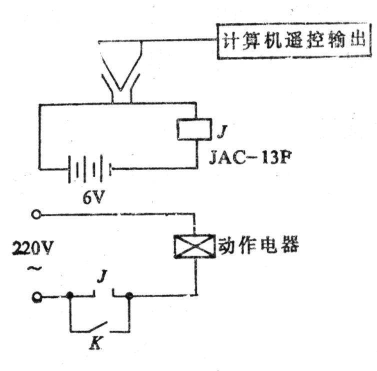


图 4-7 计算机遥控输出电路

具体设计用BASIC 语言来实现，程序清单见程序4-13。

程序4-13

```
5   OPN"LPRT"
10  WAIT 1
15  INPUT X
20  DIM V(X),M(X):FOR V=1 TO X:READ V(V):NEXT
    V
30  INPUT M:FOR K=1 TO X
40  IF V(K) < > M THEN 60
50  FOR J=1 TO X:IF K=X THEN LET K=1
55  M(J)=V(K):K=K+1:NEXT J
60  NEXT K
70  FOR V1=1TO 1000
80  MOTH=INT (TIME/10000) :DIY=INT (TIME/100) -
    MOTH *100:FOR V2=1 TO X
90  N=(TIME*10000-(INT(TIME/100))*1000000)/10000*
```

```
    IF N=M(V2) THEN 130
110 PRINT  "1986";MOTH;DIY;N
120 GOTO 90
130 PRINT #2:BEEP2, 3, 2:PRINT  #2:BEEP2, 3, 2:L
    PRINT N
140 NEXT V2:NEXT V1
150 DATA 6.30,6.40,7.10,7.50, 8.00, 8.10, 8.20, 9.05,
    9.15,10.00,10.20,11.05,11.15
160 DATA 12.00,12.40,14.30,14.40,15.25, 15.35, 16.20,
    16.30,17.15,18.00,18.30,19.50
170 DATA 20.00,21.30,22.00,22.30
```

执行程序时，将动作时间放入某数组中，使机内芯片上的时钟与动作时间进行比较、判断、分析和排队处理，然后看两者时间是否吻合。如两者吻合，则遥控信号发出，外部设备接受遥控将马上动作；反之遥控门关闭，无信号输出。同时计算机将取出下一个内存中的动作时间数据与中央处理机芯片上的时间进行比较。如此反复循环下去，直至全部数据处理完毕后再循环二次、三次……

在操作时将总动作次数X和需要第一次动作的值M输入计算机中，计算机能够人工智能化地从内存中找出需要的数据，然后将所有数据重新分析、排队，按先后次序编组并执行动作，操作非常方便。

第七节　扩展输出端的程序控制器

在前面介绍的实验中，我们已经讨论了APPLE-Ⅱ微型机的游戏I/O接口、软开关技术以及4个开关量输出端的特点，在实际使用中，有时需要更多的输出端，这一节将介绍

一种可使输出端增加到10个的扩展方法，根据同样原理，若使用二—十六进制译码电路，可使控制输出端扩展到16个。

一、扩展原理

我们已经学习过二进制概念以及二—十进制的转换关系。在游戏I/O接口的AN_0～AN_3四个开关量输出端中，每一个输出端都可以输出两种电平，即高电平1和低电平0，这样，四个输出端的电平将有16种组合，有时根据实际需要也常选用其中的前10种组合。这种二—十进制或者二—十六进制的转换一般都由专用的数字集成电路来完成，这种电路也称为译码器。这里选用了74LS42集成电路来完成二—十进制之间的转换，该集成电路的引脚排列次序如图4-8所示。其输入端和输出端的关系示意图如图4-9所示。

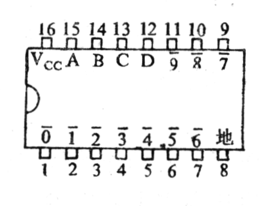


图 4-8 74LS42的引脚

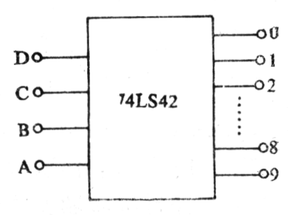


图 4-9 74LS42的输入与输出

74LS42集成电路的A、B、C、D端分别接游戏I/O接口的AN_0～AN_3端。应该注意到，74LS42是一种反相输出的译码器，即当输入相对应的电平时，在被译出的地址上由原来的高电平迅速变为低电平，因此称之为反相输出（反之，在被译出的地址上由原来的低电平迅速变为高电平的译码器称为正相输出）。译码器的输入端和输出端之间的对应电平关系如表4-2所示：

表 4-2

输入				输出									
D	C	B	A	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$	$\overline{5}$	$\overline{6}$	$\overline{7}$	$\overline{8}$	$\overline{9}$
0	0	0	0	0									
0	0	0	1		0								
0	0	1	0			0							
0	0	1	1				0						
0	1	0	0					0					
0	1	0	1						0				
0	1	1	0							0			
0	1	1	1								0		
1	0	0	0									0	
1	0	0	1										0
1	0	1	0										

由表 4-2 可知，若要在它的第 7 脚上输出控制信号，只要先找出对应的二进制数0111，然后用软开关的POKE语句，使AN_2、AN_1、AN_0端输出高电平，使AN_3端输出低电平，这在程序中是不难实现的。

二、接口电路简介

现在让我们来试一试用APPLE-Ⅱ微型机来控制10个继电器的动作。由于74LS42集成电路不能直接驱动继电器的线圈，因此在0～9的每一个输出端后面应接上功率驱动电

路，如图4-10所示：

这里选用了3DK4中功率开关晶体管作功率驱动管，74LS42集成电路的输出端分别通过电阻接至3DK4的基极，当基极输入为低电平时，3DK4截止，继电器线圈中无电流流过，它的常闭触点闭合，利用它的常闭触点控制的电器接通。举1号继电器为例，要使它的常闭触点闭合，只要使D、C、B、A的输入端分别为0、0、0、1，这相当于AN_3、AN_2、AN_1的输出分别为低电平，而AN_0端输出为高电平状态。如果要使2号继电器的常闭触点闭合，应使AN_3、AN_2、AN_0端分别输出低电平，AN_1端输出高电平。

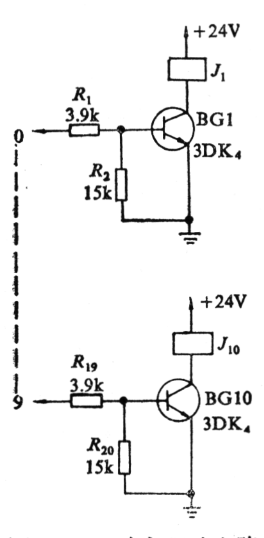


图 4-10　功率驱动电路

三、程序简介

我们先编写一个使1号继电器释放的程序，此时AN_3～AN_0的状态对应于0、0、0、1，其程序清单如下：

程序4-14

```
10  REM OPEN RELAY 1
20  POKE (—16295),1
30  POKE (—16294),0
40  POKE (—16292),0
50  POKE (—16290),0
```

```
60  GOTO 20
```

先连接好图4-10的接口电路，然后运行程序4-14，即可以看到继电器1号处于释放状态，其它9个继电器都处于吸合状态。

如果再试试2号继电器，可运行程序4-15。

程序4-15

```
10  REM OPEN RELAY 2
20  POKE (-16296),0
30  POKE (-16293),1
40  POKE (-16292),0
50  POKE (-16290),0
60  GOTO 20
```

这时，我们可以看到除了2号继电器处于释放状态外，其它继电器都处于吸合状态。

在了解单个继电器的控制原理之后，再学习多个继电器的程序控制就较为容易了。最易理解的是，在上述单个继电器的控制程序中，加上延时语句后再连接起来，便能实现对多个继电器进行程序控制，很明显这样的程序很长，并不符合程序优化的要求。前面刚介绍过的译码器原理仍然是这里进行程序设计的关键所在。根据表4-2中输入与输出之间的逻辑关系来编写10个继电器的控制程序就方便多了。程序框图如图4-11所示，程序清单见程序4-16。

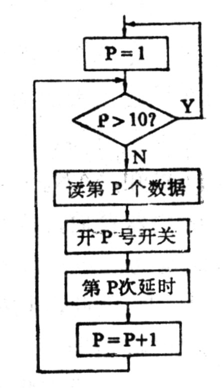


图 4-11　控制10个继电器的程序框图

程序4-16

```
3   REM MULIPLEXER CHANNEL CONTRO
    LL
5   FOR P=1 TO 10
10  READ A3, A2, A1, A0, N
20  POKE (-16296+A0),0
30  POKE (-16294+A1),0
40  POKE (-16292+A2),0
50  POKE (-16290+A3),0
60  FOR I=1 TO N:NEXT I
70  NEXT P
80  RESTORE
90  GOTO 5
100 DATA 0,0,0,0,60,0,0,0,1,120,0,0,1,0,180,0,0,1,
    1,240,0,1,0,0,300
110 DATA 0,1,0,1,360,0,1,1,0,420,0,1,1,1,480,1,0,0,
    0,540,1,0,0,1,600
```

为了便于理解程序4-16的10语句中A_3、A_2、A_1及A_0的作用，先举0号继电器为例。欲使0号继电器释放，74LS42的0号输出端应为低电平，由表4-2可知，应使其输入均为低电平。当A_3、A_2、A_1和A_0均为0时。由程序4-16中20～50语句可知，AN_0～AN_3端的输出均为低电平。同理，欲使1号继电器释放，需使AN_3、AN_2、AN_1端输出为低电平，而AN_0端为高电平，此时A_3、A_2和A_1的数字应取为0，而A_0应取为1。因此，只要改变A_3～A_0的数据，就可以控制相对应编号的继电器。

这里，我们将对应于10个继电器动作的二进制数按次序放在DATA数据中，用10语句和70语句所组成的循环语句来控制它的依次读出。DATA语句中的一组数据中，前面4个数

字表示对应的继电器，第5个数字表示延时的时间，可以根据实际需要灵活选用。当10个继电器的动作程序完成后。80语句将数据区的指针恢复，90语句又使程序从头开始继续执行。

第八节　发光二极管的控制

一、发光二极管简介

在二极管系列中有一种特殊的管子，只要在它的二端加上合适的正向电压，就能发出红光或绿光，这就是目前已得到广泛使用的发光二极管，通常称为LED。利用其发光特性，可以用来模拟彩灯控制，从中可以得到微型机控制灯光的基本原理。

发光二极管的工作电压一般为2V左右。它在正向电压下（即正极接至电源的正端，负极接至电源的负端，当发光二极管流过的电流为10mA～20mA时，它就能发出艳丽的红光或者绿光。在实际使用中，一般的电源电压都高于2V，必须加上限流电阻方可使用，否则很容易损坏它。限流电阻的阻值可根据欧姆定律计算出来。如果选用的电源电压是5V，发光二极管的工作电流为10mA，此时限流电阻应选用300Ω。如图4-12所示。

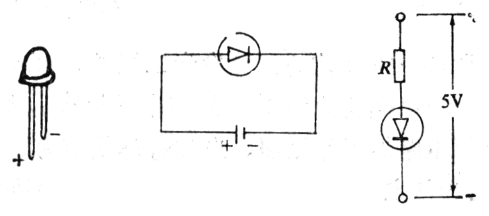


图 4-12　发光二极管及其连接电路

二、接口电路简介

1．反相器：在介绍接口电路之前，有必要先了解什么是反相器。反相器就是当输入为高电平时其输出为低电平，而当输入为低电平时其输出则为高电平的一种电路。使用反相器可以根据实际需要方便地实现高、低电平的相互转换。在本实验中，我们选用了C033六反相器，它是一种特殊的反相器集成电路，内部有6个互相独立的反相器。这里，我们实际使用了其中的4个，另外两个则空着。C033六反相器的内部电路结构如图4-13所示。

2．输出接口电路：这里使用的输出接口电路由反相器、发光二极管及限流电阻等组成，输出接口的电路图如图4-14所示：

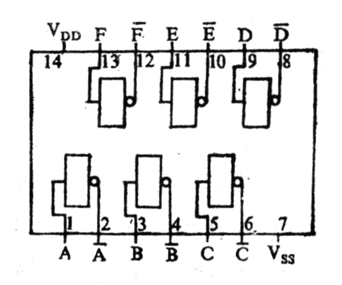


图 4-13　C033的内部电路结构

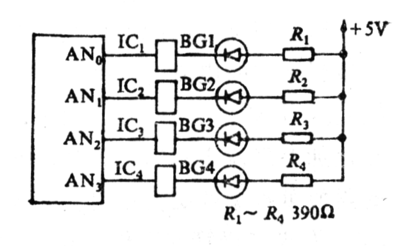


图 4-14　发光二极管控制的输出接口电路

其中，BG1～BG4均为红色发光二极管，R_1～R_4均为限流电阻，IC_1～IC_4均为C033中的四个反相器。这里以AN_0端为例，介绍电路的工作原理。当AN_0端输出为高电平时，IC_1端的输出为低电平，因而BG1得到正向工作电压而发光；反之当AN_0端的输出为低电平时，IC_1端输出为高电平，BG1

不能得到正向工作电压而熄灭。

将C033六反相器和发光二极管等器件安装在电路实验板上，然后按照图4-14接好连接线，即完成了输出接口电路。

三、控制程序简介

1. 同时发光：要使四个发光二极管同时发光，应使AN_0～AN_3四个输出端的输出均为高电平，其程序如下：

程序4-17

```
10  FOR I=0 TO 3
20  POKE (-16295+2*I),0
30  NEXT I
40  GOTO 10
50  END
```

运行后的发光二极管状态如图4-15所示：

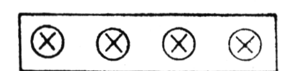


图 4-15　二极管同时发光

2. 同时熄灭：要使四个发光二极管同时熄灭，应使AN_0～AN_3四个输出端的输出均为低电平，其程序如下：

程序4-18

```
10  FOR I=0 TO 3
20  POKE (-16296+2*I),0,
30  NEXT I
40  GOTO 10
50  END
```

程序运行后的发光二极管状态如图4-16所示：

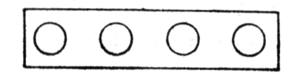


图 4-16　二极管同时熄灭

3．间隔发光：要使四个发光二极管间隔发光，应使AN_0、AN_2端输出为低电平，而使AN_1、AN_3端输出为高电平，这种状态保持一段时间后，又反过来使AN_0、AN_2端输出为高电平，AN_1、AN_3端输出为低电平，并循环反复执行。其程序框图如图4-17所示，程序清单见程序4-19。

程序4-19

```
10  FOR P=1 TO 2
20  READ A3,A2,A1,A0
30  POKE (-16296+A0),0
40  POKE (-16294+A1),0
50  POKE (-16292+A2),0
60  POKE (-16290+A3),0
70  FOR I=1 TO 750:NEXT I
80  NEXT P
90  RESTORE
100 GOTO 10
110 DATA 0,1,0,1,1,0,1,0
120 END
```

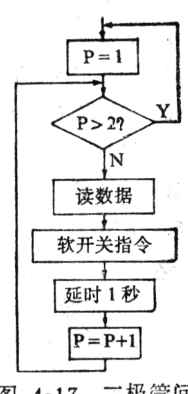


图 4-17 二极管间隔发光的程序框图

程序4-19将变量A_0、A_1、A_2、A_3分别放进30～60语句的地址中，因为这四个语句都是软开关语句。很明显，只要改变A_0～A_3的数据，就可以改变AN_0～AN_3的输出状态。这里A_0～A_3的数据用了两组，它们是0、1、0、1和1、0、1、0，比较这两组数据可以看到，0和1的位置正好相反。将这两组数据放入DATA语句中，用读数语句分别将它们读出，从而实现了发光二极管的间隔发光。70语句为软件延时，使每一种状态持续1秒左右。90语句为数据区恢复语句，在读完两次数据之后将数据区指针移到前面，为下一步的继续读取作好

准备。100语句保证了程序反复执行的连续性，使发光二极管周而复始地间隔发光。

程序运行后的发光二极管状态如图4-18所示。

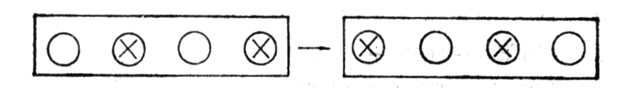


图 4-18 二极管的间隔发光

4．十个发光二极管的循环发光：在上一节中，我们已经学过游戏I/O接口的开关量输出端的扩展方法，利用本节讨论过的发光二极管的知识可以方便地将控制的发光管增加到10个。其电路图如图4-19所示。由于电路原理与前面介绍过的类同，这里不再赘述。

十个发光二极管循环发光的控制程序框图如图4-20所示。程序清单见程序4-20。

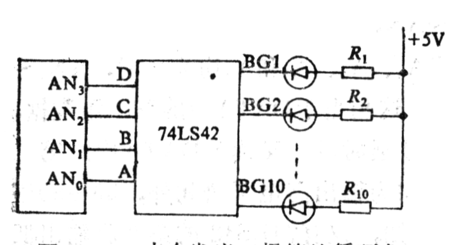


图 4-19 十个发光二极管的循环发光电路图

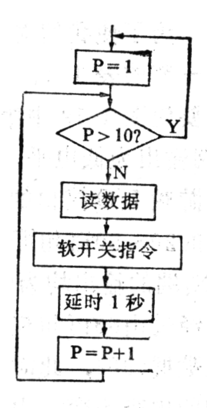


图 4-20 十个发光二极管循环发光的控制程序框图

程序4-20

```
5    FOR P=1 TO 10
10   READ A3,A2,A1,A0
20   POKE (-16296+A0),0
30   POKE (-16294+A1),0
40   POKE (-16292+A2),0
50   POKE (-16290+A3),0
60   FOR I=1 TO 750:NEXT I
70   NEXT P
80   RESTORE
90   GOTO 5
100 DATA 0,0,0,0,0,0,0,1,0,0,1,0,0,0,1,1,0,1,0,0
110 DATA 0,1,0,1,0,1,1,0,0,1,1,1,1,0,0,0,1,0,0,1
```

由于已将游戏I/O的开关量输出端通过74LS42集成电路扩展至10个，因而它的输出控制地址和方法已同上面1～3种情况有所区别。

为了使我们对该控制程序理解得更清楚，有必要再看看上一节的表4-2。该表揭示了输入与输出的对应电平关系。若要使图4-19中第一个发光管BG1发光，需使74LS42的0端输出为低电平。由表4-2可知，应使其输入为0000，即要求游戏I/O接口的AN_0～AN_3端的输出均为低电平。由程序中的20～50语句可知，当A_3、A_2、A_1和A_0均为0时，AN_0～AN_3端的输出均为低电平，这相当于DATA中的第一组数据，在经过由60语句获得的1秒延时后，又读入DATA中的第二组数据0001，此时除20语句为输出高电平语句外，30～50语句的内容不变，由表4-2可知，仅有74LS42的第2脚输出低电平，BG2管发光，其余九脚均输出高电平，所接BG管不亮。这样逐个发亮，其过程由程序中的循环语句来保证，一

直进行到第十个发光管发光，其余发光管不亮为一个过程，接着80语句将数据的指针恢复，90语句又使程序从头开始执行。这样，使得十个发光二极管循环发光。

第九节 密 码 锁

本节将介绍一种用程序来加密的电脑密码锁，该锁的开启电路输入端接至APPLE-Ⅱ微型机的游戏 I/O接口上，操作者从键盘上输入一组数据后，计算机将输入的数据同所存的密码相比较，若两者相符，程序即通过软开关语句使游戏 I/O 接口的相应输出端输出高电平，通过控制电路来开启电动密码锁。这种锁不同于一般的密码锁，由于程序的编制有很大的灵活性，我们既可以通过增加程序中设置的密码数字位数来增加开启的难度，也可以将程序设计成多级破译方式，只有依次将每一级破译后才能将锁开启。

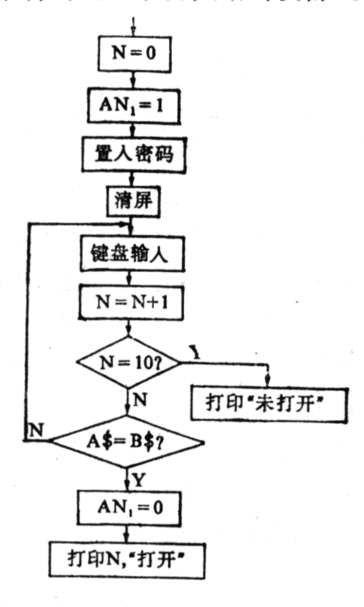


图 4-21 密码锁的程序框图

一、程序简介

密码锁的程序框图如图4-21所示。其程序清单如下

程序4-21

```
5    N=0
```

```
10  POKE  -16293,0
15  PRINT  "INPUT THE LOCK NUMBERS"
20  INPUT A$
30  CALL  -936
40  FOR I=1 TO 2000:NEXT I
50  PRINT  "ENTER THE NUMBERS"
55  INPUT B$
60  N=N+1
70  IF N=10 THEN 110
75  IF A$<>B$ THEN 55
80  POKE  -16294,1
90  PRINT  "N=";N;"     ";"OPEN"
100 GOTO 100
110 PRINT  "NO OPEN";GOTO 110
120 END
```

程序中 5 语句将统计开锁次数的N计数器清零，10 语句为软开关语句，使AN1端输出高电平，通过接口电路将电锁通电锁上，20语句是从键盘上输入密码，30语句清除屏幕，55语句是从键盘上输入所猜测的密码，由75语句可知，若密码猜对时，即执行80语句，使AN1端输出低电平，通过接口电路使电锁失电而释放。若密码未猜对，程序转至55语句，需重新输入一次，程序继续执行，在这个过程中，每输入一次密码，由60语句统计一次，当输入次数达到10次时，程序转至110语句，进入死循环，如不重新启动程序，锁就再也不能打开了。

二、输出接口及执行电路

图4-22所示是输出控制接口电路，这里采用了可自制的电磁栓做受控电锁。图4-23所示是电磁栓电路。

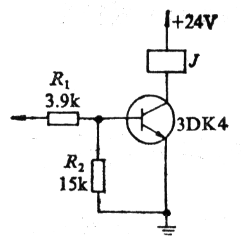


图 4-22 输出控制接口

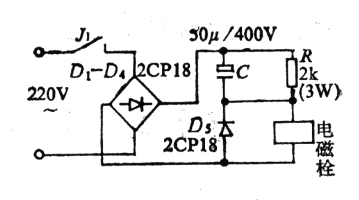


图 4-23 电磁栓电路

当游戏 I/O 接口的第14脚（AN_1端）输出为低电平时，三极管3DK4截止，继电器J释放，因为它的常开触点 J_1 串接在电磁栓的电源回路中，所以电磁栓的电源断开，电磁栓释放；当第14脚输出为高电平，三极管3DK4 导通，继电器J吸合，J_1 闭合，220V 交流经过整流后通过R加至电磁栓的绕组，R 与绕组电阻组成分压电路，C 起加速电容的作用。当电源接通瞬间，因C 两端电压不能突变，所以绕组两端的电压最高，以保证电磁栓动作。而长时间工作时，因C 两端的电压经一定时间后即充电到R 与绕组电阻的分压值，所以绕组工作电压较低，这样不会引起电磁栓过热。电磁栓的结构如图4-24所示。

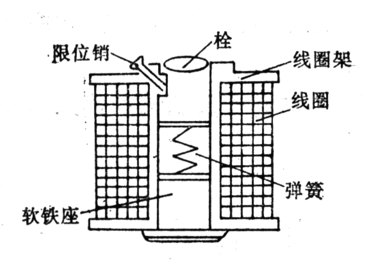


图 4-24 电磁栓的结构

这里，栓的斜面斜度为 60°，栓的最大高度为 25mm，栓上开的键槽深、宽各为 2.5mm。当线圈里有直流电流通过时，产生磁场，栓在磁力作用下被吸进线圈架，断电后，栓在复位弹簧的作用下弹出。栓上的键槽和限位销起的作用是防止栓在复位弹簧的作用下弹出线圈架。

电磁栓的线圈架用塑料制成，弹簧用直径为 1mm 左右

的钢丝绕成，其直径为8mm左右，线圈可用线径为0.19mm的漆包线绕满线圈架为止。

第十节 开关状态的读入实验

我们在第二章中已经学习过APPLE-Ⅱ微型机的游戏I/O接口，它的PB_0～PB_2端分别为一位状态输入端，用PEEK（地址）语句可以将其对应端的状态读入计算机。PB_0、PB_1、PB_2端的对应地址分别为－16287、－16286、及－16285。本节将介绍开关状态的读入实验。

一、单端开关状态的读入

用两根导线分别接至游戏I/O接口上的PB_0端（第2脚）接地端（第8脚）上，导线的另外两端接至按钮开关K上，如图4-25所示：

单端开关状态的读入的程序框图如图4-26所示。

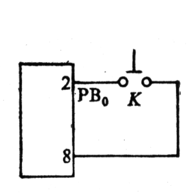


图 4-25 单端开关状态的读入

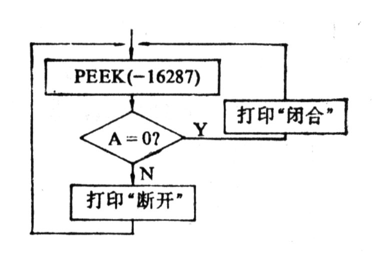


图 4-26 单端开关状态读入程序框图

其程序清单如下：

程序4-22

```
10  A=PEEK (−16287)
```

```
20  IF A=0  THEN 50
30  PRINT  "OPEN"
40  GOTO 10
50  PRINT  "CLOSE"
60  GOTO 10
```

程序 4-22 运行以后，按钮尚未按下时，显示屏上出现"OPEN"；当按钮按下后，显示屏上出现"CLOSE"，由此可知，按钮开关K的状态已经被微型机检测出来。

现在再让我们试试PB_1端的检测，将插入PB_0端（第2脚）的导线换插入PB_1端（第3脚上），其它接法不变，然后重复上述实验过程，发现当按钮按下时，显示屏上没有任何反应，因为原来程序中的地址 -16287 对应于PB_0端，所以应把地址改为 -16286，然后运行程序并变换按钮状态，便可得到上述过程的相同结果。

二、开关状态的巡测

如图 4-27所示，在PB_0、PB_1、PB_2三个开关量输入端上分别引出三根导线，通过三个按钮开关接至地端，这三个开关分别为K_0、K_1和K_2。

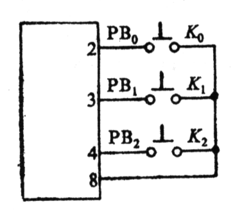


图 4-27 开关状态的巡测

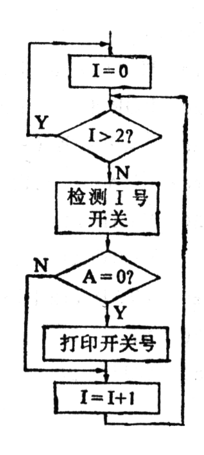


图 4-28 开关状态巡测的程序框图

三个开关状态巡测的程序框图如图4-28所示。其程序清

单如下：

程序4-23

```
10  FOR I=0 TO  2
20  A=PEEK (-16287+I)
30  IF A=0 THEN PRINT  "SWITCH";I;"IS ON"
40  NEXT I
50  GOTO 10
```

程序运行后，在显示屏上似乎没有什么反应，当按下K_0按钮时，显示屏上出现“SWITCH 0 IS ON”。再试试另外两个按钮，当按下K_2按钮时，显示屏上出现了“SWITCH 2 IS ON”。由此可知，$K_0 \sim K_2$的开关状态已经通过$PB_0 \sim PB_2$端被计算机读入。至于显示屏上的输出显示格式，只要在程序上作一些修改便可。试运行程序4-23，并与程序4-24加以比较。

程序4-24

```
10  HOME:TEXT
20  FOR I=0  TO 2
30  A=PEEK (-16287+I)
40  IF A=0 THEN 60
50  VTAB (12+I * 2):PRINT "SWITCH"; I; "IS OFF":
    GOTO 70
60  VTAB (12+I*2):PRINT "SWITCH";I;"IS ON"
70  NEXT I
80  GOTO 20
```

第十一节　智能计数器

在工业生产上，计算半成品数、产品的累计、某设备或者

某一装置的工作次数等统计问题，如果使用电子计算机来控制，不仅能够轻而易举地完成计数任务，同时还能对被测对象进行数值预置，而且电子计算机还有数据处理存储、打印输出等功能，可以组成智能型的计算装置。用微型机控制的这种智能计数器，只需在微型机上加上简单的输入接口电路即可。

一、接口电路简介

这里，接口电路的任务是将被测对象的数量转变成计算机能够接受的开关信号，然后输入电子计算机后进行处理。

在自动控制中，光电技术是用得很多的。在光电计数中，将被统计对象的数量转换成电信号，然后通过电子线路或数字计算机进行累加。本节介绍的光电转换电路选用了性能优良、价格便宜的 555 时基电路，是一种新颖的光电转换电路。而且，555 时基电路还具有良好的负载能力，它的最大输出电流为 150mA，可直接驱动小型继电器，然后通过继电器的触点，将信号送入计算机。

图4-29是光电接收装置的电路图。

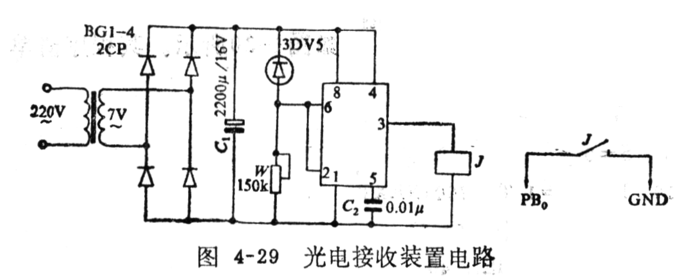


图 4-29　光电接收装置电路

这里用 3DU5 光敏三极管作受光器件。在普通光线下，3DU5的内阻约为数百kΩ。而在较强的光源照射下，其内阻

迅速下降到数kΩ，电阻值的变化相当大。我们利用3DU5的这一特性，将它接入555时基电路的输入回路中。在普通光线下，由于3DU5的数百kΩ电阻同W的电阻组成的分压器作用，所以555时基电路的第2、第6脚的电压低于三分之一电源电压。根据555电路的特性，其输出端第3脚将输出高电平（高电平接近于电源电压），此时继电器J吸合。当3DU5受到较强的光源照射时，3DU5的电阻值迅速降到数kΩ，这时分压器的输出明显升高，使第2、6脚的电位高于三分之二电源电压，其输出端第3脚即输出低电平（低电平接近于0V），此时继电器J释放。由于继电器的常开触点接至游戏I/O接口的PB_0端（第2脚）和GND端（第8脚），就把光电检测装置的信号，通过PB_0端送入了微型机。

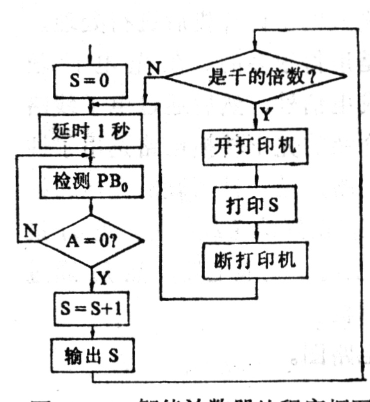


图 4-30　智能计数器的程序框图

二、程序简介

智能计数器的程序框图如图4-30所示。其程序清单如下：

程序4-25

```
5    S=0
10   FOR I=1 TO 750:NEXT
15   A=PEEK (-16287)
20   IF A=0 THEN 40
30   GOTO 15
40   S=S+1
```

```
50  PRINT  "S=";S
60  IF INT (S/1000)=S/1000 THEN 80
70  GOTO 10
80  PR#  1
90  PRINT  "S=";S
100 PR#  0
110 GOTO 10
```

在程序4-25中，5语句使累加计数器清零，15语句通过PEEK语句对PB_0端的状态进行检测，若有物体通过该光电系统时，继电器J吸合，触点闭合而使A＝0，于是程序转向40语句而得到计数累加，并由50语句输出每一次的累加数，60语句用来判断累加数是否需要在打印机上打印出来，这里选定为1000次打印一次，这个次数可以根据实际需要自己设定。如果是没有打印或打印结束，程序将转向10语句，该语句为软件延时。应该注意到，合理选取延时时间是程序的关键之处。若该延时时间加上程序运行所耗的时间(这里，程序运行时间可忽略不计)短于继电器J的接点处于闭合状态的时间，则会出现在下一次执行15语句时，PB_0的状态仍然是零，继续进行累加，显然这是误计数。但是该延时时间也不宜过长。如果过长，便把下一次的继电器动作过程漏检，也会引起误计数。因此该延时时长的选择原则是：既要保证略长于一次继电器的吸合时间，又要短于两个相邻被测物通过光电系统的时间。程序中30语句的作用是，当没有物体通过光电系统时，程序一直处于PB_0的检测状态。

三、制作要点

如用电路实验板完成本实验时，电源部分可用成品稳压

电源，如需将接口电路部分组装成一个独立的装置，则需要制作电源部分，可以选用普通晶体管收音机稳压电源使用的小型电源变压器，次级电压为7V 左右即可。由于555时基电路有着很宽的工作电压范围（4.5V～18V都能正常工作），所以它的电源部分不需要经过稳压处理。

在制作前，应对光敏三极管 3DU5 作一下测试。先在普通光线下用万用表的 kΩ 档测其内阻，然后再用较强的光源，如灯泡，对照3DU5 的窗口，再测其内阻，便可以判断其性能好坏。

这里选用了JRX-13F型小型继电器，也可以使用性能接近的继电器，其它器件均无特殊要求。

第十二节　测 量 时 间

在前面介绍的实验中，我们已经学过用APPLE-Ⅱ微型机进行定时控制的方法。实际上，微型机不但可以输出定时脉冲，还可以用于时间测量。在过程控制中，有时需要定时信号，有时则需要对受控对象进行时间测量，然后根据测量的结果及要求进行比较并作出判断，再发出相应的处理指令。定时和测时技术在微型机应用中都是极为常见的。

一、微型机测时的原理和特点

这里介绍的测量时间的方法，仍使用了APPLE-Ⅱ微型机的游戏I/O 接口。先将被测对象接至接口的开关状态输入端，然后由微型机对开关状态进行检测，如需测量开关的闭合时间，先检测开关是否闭合，若未闭合，则回到等待状态；若已闭合则进入软件延时和计时。每隔一定的时间再对

开关状态进行检测，判断开关是否断开。若未断开，继续进入延时、计时；若断开则输出所测的时间。显然，测时的精度同采样的间隔时间有关。间隔时间越短，其精度越高；间隔时间越长，其精度越低。

和一般的机械和电子测量时间的方法相比，微型机测时的最大特点在于：它能通过软件中的预置要求，在测时过程中进行数据处理和储存；能根据测时结果自动发出相应的控制指令；在自动复零、多路控制等方面也比传统的机械或电子测时方法更为方便。

现在让我们自己动手来完成下面两个测时实验。

二、开关闭合时间的测量

这里先做一个简单的开关闭合时间测量实验。

准备一个普通的电源开关，将开关的一端用导线接至游戏I/O接口的 PB_0 端（第 2 脚），开关的另一端用导线接至游戏I/O接口的GND 端（第 8 脚）。该连接示意图如图4-31所示。注意开关与导线的连接处需要用电烙铁焊接，导线与接口的接触也必须良好。

测量开关闭合时间的框图如图4-32所示。其程序清单见程序4-26：

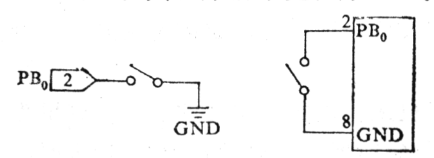


图 4-31 开关闭合时间的测量

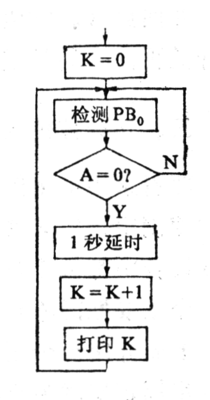


图 4-32 开关闭合时间的测量程序框图

程序4-26

```
5   REM MEASURE TIME PROGRAMMING
10  K=0
20  A=PEEK (-16287)
30  IF A<>0 THEN 20
40  FOR I=1 TO 750:NEXT
50  K=K+1
70  PRINT  "K=";K;"S"
75  GOTO 20
80  END
```

程序中10语句先将计时器清零。20语句为开关状态读出语句。30语句用作判断开关状态的条件。若开关未闭合则回到20语句；若开关闭合、即A＝0则执行40语句。该语句为软件延时，得到1秒的时延，并在50语句中得到计时累加。开关闭合一段时间后又断开时，其闭合时间由70语句打印出来，并回到10、20语句，处于PB_0状态检测阶段，一直等待到开关再闭合后又进行累加计时。

三、定时器的测试

随着家用电器的普及，定时器已进入千家万户。在工业自动控制中，定时器的应用也极为广泛，在多数自动控制的电器设备中都装有各种精度的定时装置。定时精度是定时器的一个主要性能指标。

下面介绍一个用微型机来检测定时精度的实验。这里我们选用标定精度为±5%的定时器。如果用它的30秒定时档做实验，根据±5%计算，它的定时范围应在27～33秒之间才能算合格，否则为不合格。

检测时应把定时器的两根输出控制线分别接至微型机游戏I/O接口的PB_0端（第2脚）和接地端（第8脚）。

下面介绍测时程序。其程序框图如图4-33所示，其程序清单见程序4-27

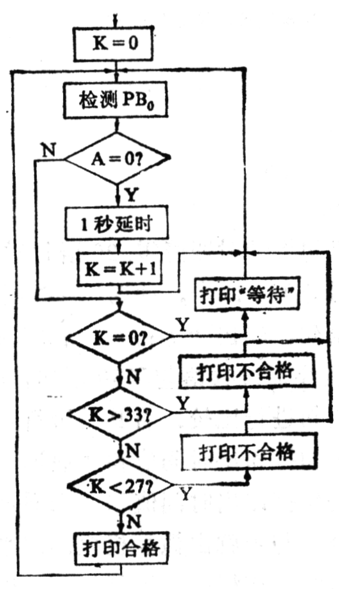


图 4-33 测量定时器时间的程序框图

程序4-27

```
5    REM TEST TIMER PROGRAMMING
10   K=0
20   A=PEEK (-16287)
30   IF A<>0 THEN 60
40   FOR I=1 TO 750:NEXT
50   K=K+1
55   GOTO 20
60   IF K=0 THEN 150
```

```
70  IF K>33 THEN 110
80  IF K<27 THEN 130
90  PRINT  "K=";K;"PASS"
100 GOTO 20
110 PRINT  "K=";K;"NO PASS"
120 GOTO 20
130 PRINT  "K=";K;"NO PASS"
140 GOTO 20
150 PRINT  "WAIT"
160 GOTO 20
```

程序中10语句将计时器清零。20语句对 PB_0 端进行状态检测。若开始时开关未闭合，此时K＝0，则程序转向60语句后又转到150语句，输出“等待”；若定时器已进入定时阶段，即A＝0，程序通过40语句和50语句计时。定时过程结束时，由60～80语句进行比较，看其是否符合测时误差要求。若合格则由90语句打印出时间值和“通过”，若不合格则由110语句或130语句打印出时间值和“未通过”。这样，定时器的定时精度合格与否就由微型机检测出来了。

为了方便检测，在程序中还可加入相应的发声语句和作图语句。例如，当检测到不合格产品时，如属正超差范围，让微型机发出频率较高的声音，如属负超差范围，则让微型机发出频率较低的声音。

第十三节　电冰箱开停时间比的测试

当我们选购一台电冰箱时，总是希望它的耗电量要小一些。影响电冰箱耗电量的因素较多，如整个冰箱系统的设计是否合理，压缩机的制冷量是否足够，发泡保温材料是否致

密等。那么，怎样判断它的电耗及其有关性能呢？

首先让我们了解一下电冰箱的工作过程。整个过程分为两个阶段：一个是开机阶段，即压缩机运转阶段；另一个是停机阶段，即压缩机停止运转阶段。在满足同样制冷要求的条件下，我们总是希望它的开机时间短一些，而停机时间长一些，因此压缩机的开停时间比，即它的工作系数是衡量电冰箱的一个重要技术指标。由于电冰箱的运行时间较长，加上压缩机的开与停，人们不易觉察出来，如用人工检测是比较麻烦的。本节介绍一种使用 APPLE-Ⅱ 微型机进行上述检测的方法。

一、压缩机开停状态的检测原理

怎样将电冰箱的开停状态输入计算机？我们首先研究一下电冰箱开机和停机时的状态特点。当电冰箱处于开机状态时，约有 0.7A～1A 左右的电流通过压缩机的线圈，刚开始的起动电流还要大一些。经过一段时间运行后，冷藏室的温度逐渐下降到温控器的动作点，于是温控器的触点将压缩机从电路中断开，此时电路中没有电流通过。要区分这两种不同的状态，比较常用的方法是，将电流继电器串接在压缩机的电源回路中。这里需要注意的是，必须根据实验所用电冰箱的实际工作电流，来选用电流继电器。多数电流继电器的动作电流是可以调节的，应预先调到它的吸合电流点上。当电冰箱的压缩机运转时，电流继电器吸合；当压缩机停止时，电流继电器没有电流通过而释

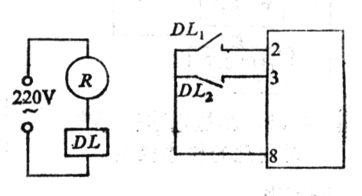


图 4-34 压缩机开停状态的检测电路

放。我们把电流继电器的两组触点（常开和常闭触点）分别接至APPLE-Ⅱ微型机的游戏I/O插座上，然后输入一段检测程序，就能让微型机来代替人工测试了。图 4-34 是测试电路图。

在图4-34中，*R*表示电冰箱的压缩机，*DL* 表示所用的电流型继电器线圈，220V交流电压经过电流继电器*DL* 加至压缩机，DL_1和DL_2分别为电流继电器的常开和常闭触点，它的中间触点接至游戏I/O接口的第 8 脚（接地端），当压缩机运行时，*DL*吸合，它的常开触点DL_1闭合，此时2脚PB_0与地端接通；当压缩机停止运行时，*DL* 释放，它的常闭触点DL_2与第 8 脚接通，此时 3 脚PB_1端与地端接通，同时，2脚PB_0与地端断开。这样，压缩机的状态通过游戏接口的PB_0和PB_1端送入计算机。

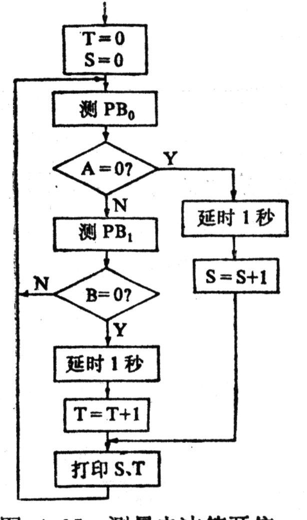


图 4-35 测量电冰箱开停时间比的程序框图

在编制程序和调试阶段，也可以用三根导线和一个双掷开关来模拟上述过程，然后再接入实际电路。

二、测试程序简介

测量电冰箱开停时间比的程序框图如图4-35所示，其程序清单如下：

程序4-28

```
5    REM TEST REFRIGERATORY TIME
8    T=0:S=0
```

```
10    A=PEEK (-16287)
20    IF A=0 THEN 50
25    B=PEEK (-16286)
28    IF B=0 THEN 30
29    GOTO 10
30    FOR I=1 TO 750:NEXT
33    T=T+1
40    GOTO 70
50    FOR I=1 TO 750:NEXT
60    S=S+1
70    PRINT "S=";S;"T=";T;"  V= ";S/(S+T)
75    GOTO 10
RUN
S=1   T=0    V=1
S=2   T=0    V=1
S=3   T=0    V=1
S=4   T=0    V=1
S=5   T=0    V=1
S=6   T=0    V=1
S=7   T=0    V=1
S=8   T=0    V=1
S=9   T=0    V=1
S=10  T=0    V=1
S=10  T=1    V=.909090909
S=10  T=2    V=.833333333
S=10  T=3    V=.769230769
S=10  T=4    V=.714285714
S=10  T=5    V=.666666667
S=10  T=6    V=.625
```

S=10　T=7　　V= .588235294

S=10　T=8　　V= .555555556

S=10　T=9　　V= .526315789

S=10　T=10　V= .5

程序中分别用S和T表示压缩机的开、停时间，8 语句将T和S清零。10语句和20语句用来检测压缩机的开停状态，若压缩机处于运行状态，PB_0与地接通，这时A = 0，则由 20 语句转入50语句，在该语句中得到1秒的延时，然后由 60 语句累加起来；若压缩机处于停止运行状态，这时A≠0，则由20语句向下，执行下一语句即25语句。25语句和28语句也是用来检测压缩机状态的，当A≠0时，由于DL_1和DL_2是同一个继电器的两组反向触点，所以B = 0，则由28语句转入30 语句，并在该语句中得到1秒延时，然后由33 语句累加起来。无论是开机时间还是停机时间，每检测一次后都转入70语句，该语句把开时间S和停时间T分别打印出来，同时用式子V = S/(S + T)计算压缩机运行时间占冰箱使用总时间的比例（即工作系数），然后打印出来。照此继续进行下一次检测。压缩机的整个运行和停机过程就这样被计算机既准确而又直观地显示出来。很明显，V的值越小，电冰箱的相对电耗越小。

程序4-28是为了实验调试和演示方便而编制的，因此每秒打印一次，实际上并不需要这么多数据，读者可以根据实际需要修改。在程序中还可以加入统计整个运行过程中压缩机开停次数的语句，使程序的功能更强。

第十四节　测量电阻

电阻是最常见的电子元件之一，几乎所有电子线路都离

不开它。测量电阻阻值大小，最常用的方法是使用万用表的欧姆档，在一些要求较高的场合，也可以用数字式仪器来测量。数字式电阻仪的精度一般比指针式万用表来得高，当然其价格也贵得多。

利用微型机测量电阻，不仅测量精度较高，而且能够自动地显示、打印、存盘，可作为一种智能化电阻测试仪。一般来说，只要给微型机配上一个专用接口电路，编制一个测试软件即可。对于APPLE-Ⅱ微型机来说，测量2kΩ～150kΩ范围内的电阻更为方便。在该微型机的硬件设计时已经考虑到这一点，采用了一个特殊的简易A/D转换电路。在软件方面，它的监控程序中已经有一段可以用来测量电阻的软件，只要将上述范围内的电阻，直接接到游戏I/O接口的相应端上，再输入一段用BASIC语言编写的程序，就可以测其阻值。在该电阻范围以外的电阻，也可以通过游戏I/O接口测量，但需外加接口电路。下面我们对测量原理、硬件电路以及测试软件分别加以介绍。

一、基本测量原理

电阻测量原理一般都基于欧姆定律。当加在电阻上的电压为一恒定值时，根据欧姆定律 $I=V/R$，只要测出通过该电阻的电流I，即可得到R的值。万用表的Ω挡就是利用了这一原理。

电阻的测量也可以利用电容充电和放电原理。如图4-36所示，电源电压通过电阻R向电容C充电，这里电源电压为一固定值，当电阻R固定时，电容C的容量越大，A点电压升到某一定值的时间越长；当电容C固定时，电阻R的值越大，A点电压达到某一定值的时间也越长，即时间 $T=K\cdot R\cdot C$，当

K和C为一定值时，只要测出时间T，即可求出R的值（此处假定电容器的漏电流和电源的内阻都非常小，可以忽略不计）。

下面我们将介绍用555时基电路设计的接口电路，其中电容C是固定的，而达到某一固定电压的时间T将随接入电阻R的变化而变化，利用前面学过的微型机测量时间的方法将时间T测出，然后再换算成电阻。为了进一步了解其中的转换原理，有必要先介绍一下555时基电路。

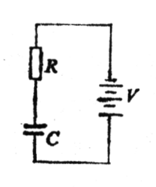


图 4-36 电容的充放电

二、555时基电路简介

555时基电路是一种模拟、数字混合式双极型集成电路，用它组成的时间电路具有电路简单、定时精度高、温度漂移小、电源电压范围宽、功能灵活、负载能力强等优点。用555时基电路再加上少量的外接元件，可以设计出多种新颖电路，它是一种应用极为广泛的集成电路。

555时基电路的封装形式有两种。一种是8脚圆形的T型封装，另一种是8脚双列式V型封装，如图4-37所示。两种封装形式的功能完全相同。目前一般都用V型封装，T型封装已很少见到。555时基电路的引脚功能见表4-3，它的输入端与输出端之间的关系见表4-4。了解这些功能、弄清楚这些关系，对于学习和设计电路是十分有益的。

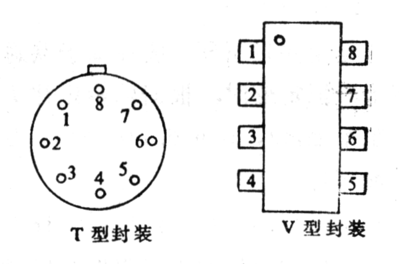


图 4-37 555时基电路的封装形式

表 4-3

引脚号	1	2	3	4	5	6	7	8
引脚名	GND	$\overline{V}_{TR}$	OUT	$\overline{R}_{D}$	V_c	V_{TH}	SW	V_{cc}
功　能	地	触发端	输出端	复位端	外接控制电压端	阀值端	放电端	电源端

表 4-4

$\overline{R}_{D}$	V_{TH}	$\overline{V}_{TR}$	OUT
	$>\frac{2}{3}V_{cc}$或$>V_c$	$>\frac{1}{3}V_{cc}$或$>\frac{1}{2}V_c$	0
1	任　意	$<\frac{1}{3}V_{cc}$或$<\frac{1}{2}V_c$	1
	$<\frac{}{3}V_{cc}$或$<V_c$	$>\frac{1}{3}V_{cc}$或$>\frac{1}{2}V_c$	保　持
0	任　意	任　意	0

这里主要介绍555时基电路的定时功能，如图4-38所示。电路中的外接元件只用了三个。先讨论一下电阻R和电容C与第2、6脚连接处的A点电位变化过程和输出端第3脚的电位关系。

根据555时基电路的真值表，当A点的电位低于三分之一电源电压且第4脚接正电源时，555时基电路的第3脚将输出高电平1（实际输出电压稍低于电源电压）；当A点的电位高于三分之二电源电压时，3脚将输出低电平0（实际输出电压稍高于0电压）。

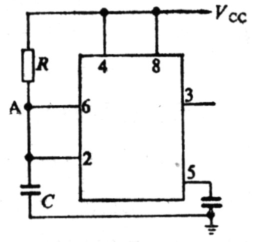


图 4-38　555时基电路的定时功能电路

当电源接通时，由于电容C两端的电压不能突变，电源电压通过电阻R对电容C充电，A点的电位将逐渐上升，上升

到一定时间T时，A点电位将达到三分之二电源电压，这时555时基电路的第3脚即输出低电平。很明显，当电源电压和电容C都为定值时，A点电压达到三分之二电源电压的时间T与电阻R的值成正比。它们之间的关系还可以用数学方法计算出来。因为

$$2/3\ V_{cc}=V_{cc}\ (1-e^{-t/RC})$$

$$t=-R\cdot C\cdot ln\ (1/3)$$

所以$t\approx1.1R\cdot C$（s），式中R的单位为Ω，C的单位为F。

这样，当电路中的R为未知值时，只要用微型机测出T值，就可以计算出R的阻值。当然，这种计算都由微型机在瞬间自动完成。

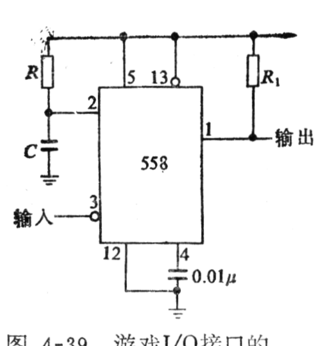
我们知道，电容充电的过程是个连续变化的过程，是个连续量，而微型机测量时间是用计数的方法来实现，它统计出来的是数字量，这个T值测定过程实则上是一个将模拟量转换成数字量的过程，我们把这个过程称为模数转换或A/D转换。

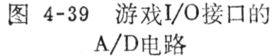
有了以上预备知识，再来分析APPLE-Ⅱ微型机中的具体电路就比较容易了。

三、特殊的A/D接口电路

前面我们已经简单地介绍过，APPLE-Ⅱ游戏I/O接口中的GC_0～GC_3端是为模拟量的输入而设计的。图4-39是APPLE-Ⅱ微型机的实际电路中的有关电路。图4-40是接脚的波形图。

实际电路使用了定时器组件558集成电路，它的内部由四个独立的555时基电路所组成，图4-39所示仅是其四分之一。每一部分都作为一个A/D转换接口。由此可知，这个四




图 4-39 游戏I/O接口的 A/D电路

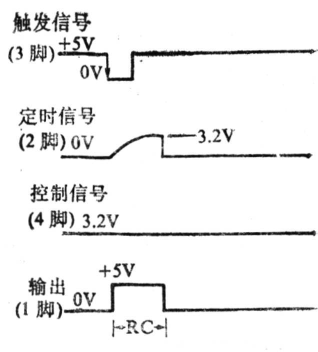


图 4-40 接脚的波形图

路输入的特殊A/D接口电路成本是很低的。

当我们在它的触发输入端（第 3 脚）送入一个负向脉冲时，555 时基电路的内部有一个内接晶体管立即使它的定时输入端第2 脚旁路到地，此时原来为 0V 的1脚输出变为高电平。负脉冲瞬间消失后，它的第2脚对地断开，5V 电源电压通过R对电容C充电，2 脚电压逐渐上升，当超过三分之二电源电压时，时基电路迅速翻转，1脚输出变为低电平。1端输出高电平的持续时间（即图4-40中的脉宽）取决于RC 的乘积。由于电容C的容量已固定为0.02μF，所以只要测出它的脉宽，再经过算式$T=1.1RC$换算，就可以得到R的阻值。

四、T值的测量

这里，测量时间T 的方法是：先用程序触发 555 时基电路，使其输出为高电平，此时微型机即开始计数，当电容 C 上的充电电压达到三分之二电源电压时，555 时基电路的输出即转为低电平，这时微型机计数结束，并输出结果。

在APPLE-Ⅱ微型机的监控程序中，有一段测量延时 T

的子程序，程序清单如下：

程序4-29

```
FB1E—  AD 70 C0      LDA   $C070
FB21—  A0 00         LDY   #$00
FB23—  EA            NOP
FB24—  EA            NOP
FB25—  BD 64 C0      LDA   $C064,X
FB28—  10 04         BPL   $FB2E
FB2A—  C8            INY
FB2B—  D0 F8         BNE   $FB25
FB2D—  88            DEY
FB2E—  60            RTS
```

当程序访问地址单元＄FB1E时，首先触发555时基电路，使图4-39中的第1脚输出高电平，然后将作为计数器使用的Y寄存器初始化为0，监控程序中接着延时约4μS，再读出定时器的输出（图4-39中的第1脚）。若初次读出时，输出为低电平，则表明脉冲甚短，即电阻甚小，以Y＝0的方式退出监控程序；若初次读出时输出不为低电平，则监控程序使计数器（Y寄存器）加1，随后检查计数器是否溢出为0。若计数器溢出，意即Y寄存器从255溢出为0；如未溢出，监控程序再次读输出，循环时间为11个时钟周期，共10.8μS。将计数器的计数结果乘以10.8μS即可得到实际时间T。当定时器输出变为低电平时，即退出监控程序，并输出结果。

有了以上这段测时子程序，就能直接用BASIC语言来编写读出游戏I/O接口中的4个模拟量输入端GC_0～GC_3的程序了。具体的方法是将对应的输入端赋值给某一个变量，例如要读入GC_0端，应使用LET　Y＝PDL（0）语句，读入GC_1端，应使用LET　Y＝PDL（1）语句等。BASIC语言编写

的程序执行到这里时，即可自动进入上述的监控程序。这对只学过BASIC语言但没有学过汇编语言的初学者来说，无疑带来了很大的方便。

五、用BASIC语言编写的电阻测量程序

有了上述专用PDL语句后，用BASIC语言编写的电阻值测量程序显得十分简练，其程序框图如图4-41所示。其程序清单见程序4-30：

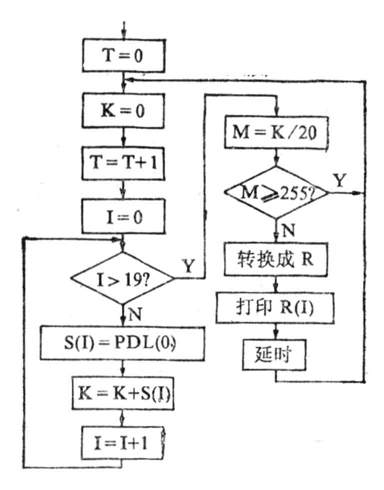


图 4-41 电阻测量的程序框图

程序4-30

```
5   REM MEASURE RESISTANCE
10  DIM S (20)
20  T=0
30  K=0
35  T=T+1
40  FOR I=0 TO 19
```

```
50  S(I)=PDL(0)
60  K=K+S(I)
70  NEXT I
80  M=K/20
90  IF M>=255 THEN 30
100 R=((M*10.8)/(1.1*0.022))/1000
110 PRINT
120 PRINT
130 PRINT  "R(";T;")=";R;"K"
135 FOR I=1 TO 3000;NEXT
140 GOTO 30
```

程序4-30中，50语句用S（I）=PDL（0）使程序进入测时子程序。我们采用的是累计次数的方法，S（I）实际上是测到的累计次数。为了减少测量误差，在40语句至70语句之间的循环体内，用循环语句将S（I）测试20次，并在60语句中得到累计，然后在80语句中求出平均值。在90语句所设的条件中，放置了八位二进制的最大数255，当被测的电阻值超出所允许的测量范围（这里约为100kΩ）或者未接入电阻时，M均大于255，此时程序将跳过打印输出语句，回到30语句，然后继续执行。当被测电阻的阻值在规定的测试范围内时，经90语句判断后执行100语句（M<255）。该语句为实际电阻值换算语句。因为$T=1.1RC$，所以$R=T/1.1\times C$，而T等于统计出的平均次数乘以一次所耗的时间10.8μs，所以$R=[(M\times 10.8)/(1.1\times 0.022)]/1000$，式中0.022为电容$C$的值，除以1000是为了得到电阻的kΩ单位。程序中还有统计被测电阻次数的功能。20语句将代表被测次数的T计数器清零，35语句为计数器累加语句，最后用R（T）的打印输出语句将被测次数反映出来。

被测电阻的接入方法很简单，如图4-42所示，在关机状态下，从游戏I/O插座的第1脚（+5V端）和第6脚（GC_0端）分别引出导线，导线与电阻的连接处焊上小夹子，也可焊上合适的插座，使电阻的接触可靠。

然后开启主机，输入并运行程序4-30。没有接入电阻时，因输入端处于开路状态，已超出其测量范围，显示器上没有数字输出；当接入2kΩ～100kΩ范围内的电阻时，显示器即显示出实测电阻值。上述范围以外的电阻，虽然也可以用这一原理测量，但需改换相应的电容。如果在主机电路板上拆换0.022μF电容显然是不太现实的，因此需要另外制作一个类似的接口电路，这样可以根据被测电阻的大致范围，灵活选用合适的电容。具体方法详见本章第十七节。

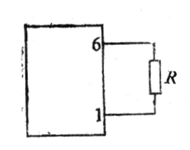


图 4-42 被测电阻的接入

第十五节 测量温度

一、温度测量的一般方法

测量温度对于我们来说并不陌生。看一下挂在墙壁上的温度计，就可以知道今天的气温是多少度；人感到有些不舒服，用体温计测一下自己的体温，这样的测温方法称作人工读数。而在工业自动控制和科学试验中更离不开温度的检测，因此人们研制成功多种测量温度的方法及仪器，并广泛

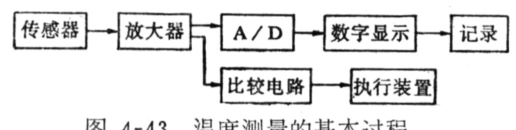


图 4-43 温度测量的基本过程

应用于各个领域。

虽然测温的方法较多，但其基本原理大致相同。图4-43是测温方法基本原理的示意图。

下面介绍这个基本过程。传感器将被测温度转变成一个微弱的电信号，然后经高增益放大器放大后，用指针式表头指示，或者送至模拟—数字转换电路（即A/D转换），将模拟量转换成数字量，并在数字显示仪表上显示出来。放大器输出的另一路信号送至比较电路，同预先设定的温度点作比较，对受控对象进行自动控制。一些要求较高的地方还装有温度记录装置，以前使用最多的是温度记录仪，目前较多地用微型机控制的打印机或绘图仪来替代记录仪。

在温度检测中，传感器是很关键的一个部件，它直接影响到检测的精度和稳定性。常用的测温传感器有铂电阻、热敏电阻、热电偶以及近年来出现的集成传感器。集成传感器在200°C以下的主要性能优于前面几种传感器。温度传感器的特点是，当被测温度变化时，它的某一主要电参数将随之发生有规律的变化。如铂电阻或热敏电阻，当温度变化时，它的电阻值有明显的变化。我们先找出电阻值同温度之间的关系，然后用微型机测量电阻值，并换算成相应的温度，在显示器上显示或用打印机打印出来。

二、热敏电阻

热敏电阻是一种应用广泛的温度器件，它具有温度敏感性好，使用方便，价格便宜等优点。热敏电阻有正温度系数和负温度系数两种规格，即它的电阻值随温度的升高而增大或随温度的升高而减小。在收音机、电视机以及其它电子线路中，根据实际需要选用它们，作为温度补偿器件。

常用热敏电阻的外形一般为圆形，它的圆面一侧标有某一温度点时的电阻值，如图4-44所示。

在本节介绍的测温实验中，我们选用了负温系数的热敏电阻。先用万用表的欧姆档来测量一下它的阻值变化。假设室温较低，在10°C左右，将热敏电阻的两端接至万用表的表笔，万用表即指示出它的阻值，然后用两个手指按住电阻，由于人体体温的作用，热敏电阻的温度升高，此时万用表的指针逐渐向右偏转，即它的电阻值逐渐减小。

1k

图 4-44 热敏电阻

三、程序简介

这里介绍的测温实验，采用了先由微型机测出热敏电阻的阻值、然后自动转换成温度的方法，所以上节介绍的电阻值测量程序（程序4-30）可移用，但需稍加补充。修改后为程序4-31。

程序4-31

```
10 DIM S(20)
20 T=0
30 K=0
35 T=T+1
40 FOR I=0 TO 19
50 S(I)=PDL(0)
60 K=K+S(I)
70 NEXT
80 M=K/20
100 R=((M*10.8)/(1.1*0.22))/1000
105 P=35-4.28*R
110 PRINT
```

```
120 PRINT  "R(";T;")=";R;"K";"P=";P;"C"
130 FOR I=1 TO 6000; NEXT I
140 GOTO 30
```

增加的105语句是将电阻值转换成温度，语句中$P=35-4.28\times R$是仅根据实验中所使用的热敏电阻数据得到的近似算式，在某一温度区间误差较小。由于各类热敏电阻的温度特性差异较大，读者可根据自己使用的热敏电阻特性建立转换关系算式。在120语句中增加了温度打印输出的功能。

四、回路温度巡测

利用 APPLE-Ⅱ微型机中游戏I/O接口的四个模拟输入端，在不需要增加自制接口电路的情况下可完成四路温度的巡回检测。例如，将四路传感器中的三路分别放入双门电冰箱的冷冻室、冷藏室，另一路用来检测环境温度，便可以用微型机来检测电冰箱在运行过程中的温度变化情况，如测量冷冻室的最低温度，冷藏室的温度以及冷却速度等。

如图4-45所示，分别在四个模拟输入端接入热敏电阻R_1、R_2、R_3、R_4，

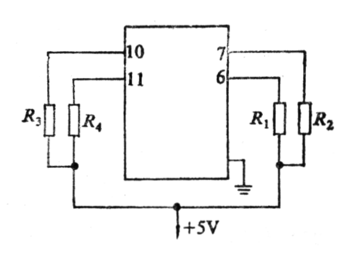


图 4-45 四路温度巡测电路

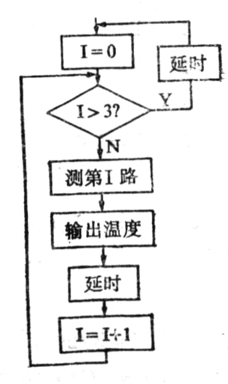


图 4-46 四路温度巡测控制程序框图

四个电阻的另一端都接到+5V端。所有的连接处都应接触良好，四个热敏电阻的温度特性应基本一致。下面简单介绍其控制程序。程序框图 如 图 4-46 所示。其程序清单如下：

程序4-32

```
5   REM MEASURE TEMPERATURE
10  FOR I=0 TO 3
20  S(I)=PDL(I)
30  PRINT  "S";I;"=";((S(I)*10.8) /(1.1*0.022))/1000;
    "K";
33  PRINT  "   T=";35-4.28*S(I);"C"
35  PRINT
38  FOR T=1 TO 100:NEXT
40  NEXT I
50  FOR I=1 TO 3000:NEXT I
55  PRINT
60  GOTO 10
```

程序4-32中的10至40语句之间的循环体用于四路温度的巡回检测。当I取0时，检测第1路，当I取1、2、3时则分别检测第2、3、4路。在一个巡测过程完成以后，用50语句中的软件延时来得到两个巡测过程之间所需的延时，改变50语句中的循环次数，即可改变巡回检测的频率。

第十六节　温度控制装置

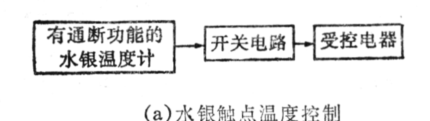
一、常见的温控装置

微型机机房需要恒温，塑料加工、金属处理、电炉炼钢等更离不开温度控制装置。温度控制涉及到相当多的领域。

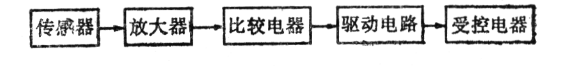
常见的温度控制装置有水银触点温度控制、传感器温度

控制等。前者用可以调节温度控制点的水银温度计作为探头，当温度达到预调点时，水银温度计短接，使同它连接的开关电路翻转，来控制受控电器。带有传感器的温控装置要复杂得多。传感器采集的微弱温度信号，经放大器放大后送至比较电路进行比较，当达到比较电平时比较器翻转，并控制受控电器。

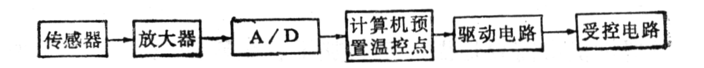
现在让我们来了解一下微型机控制温度的原理。它同样需要传感器和放大器，但不需使用比较电路。经过放大的温度信号直接送至模数（A/D）转换电路，变成数字计算机能够处理的数字量，计算机有了这个数字量后就非常方便了。我们可以将温度控制点设置在程序中，即在程序的条件语句中，将控制点的数字作为条件，同实测温度进行比较，然后发出相应的控制指令。如需改变温度控制点，只要修改一下软件中的数字。一些用前两种方法较难实现的多点温度控制、多路温度控制等，只要在计算机上编写一段程序并增加一个不复杂的接口就可以完成了。



(a)水银触点温度控制



(b)传感器温度控制



(c)计算机温度控制

图 4-47

图4-47是上述三种装置的工作原理框图，读者可以加以比较，从而进一步了解微型机控制的特点。

二、微型机温控装置

本节介绍的微型机温度控制装置，其测温部分仍然用热敏电阻作为温度传感器，并通过GC_0端送入计算机。它的温度控制点由程序中的预置数字决定。当没有达到所需恒温范围的下限温度时，加热系统工作，制冷系统关闭；当温度升过下限但未超过上限时，将维持原状态不变；一直到温度超过上限时，加热系统关闭，制冷系统开始工作。如此周而复始，使整个系统的温度恒定在某一预先指定的区间。

图4-48是模拟实验的电路图。图中R_t为热敏电阻传感器。这里分别用红色和绿色两种发光二极管来分别表示加热制冷系统的状况。两个发光管各通过一个限流电阻接地，它们的正极分别接至游戏I/O接口的AN_0端（第15脚）和AN_1端（第14脚）。实验时，根据预置的温度控制点，显示器上显示的实际温度以及发出的红

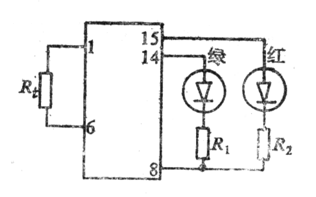


图 4-48 温控装置模拟实验电路

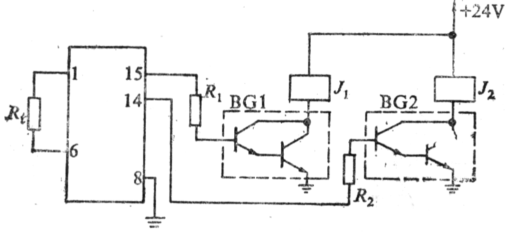


图 4-49 温控装置的实际接口电路

光或绿光，可以验证所编制程序是否正确。然后按图4-49所示接入实际控制电路。

将电路中的元件安装在电路实验板上，继电器J_1、J_2的线圈用导线接入，继电器的触点分别接至各自的控制系统。这里J_1控制加热，J_2控制制冷。功率驱动级所需的电源由外接直流稳压电源供给，BG_1和BG_2的基极分别接至AN_0端和AN_1端。当基极输入为高电平时，该晶体管导通，集电极回路中的继电器吸合；当基极输入为低电平时，该晶体管截止，集电极回路中的继电器释放。这里BG_1、BG_2选用FD30达林顿管，它是一种由复合管组成的功率器件，具有相当高的放大倍数，能用很小的输入电流控制大的驱动电流。

三、程序简介

温度控制的程序框图如图4-50所示。其程序清单见程序4-33：

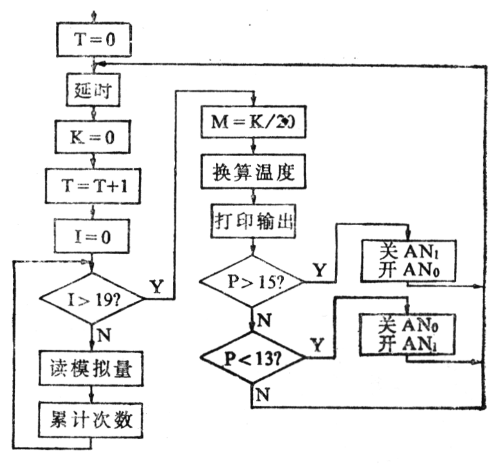


图 4-50 温度控制的程序框图

程序4-33

```
10  DIM S(20)
20  T=0
30  FOR I=1 TO 7000:NEXT I
33  K=0
35  T=T+1
40  FOR I=0 TO 19
50  S(I)=PDL(0)
60  K=K+1
70  NEXT I
80  M=K/20
100 R=((M*10.8)/(1.1*0.022))/1000
105 P=35-4.28*R
110 PRINT
120 PRINT "R(";T;")=";R;"K";"P=";P;"C"
125 IF P>15 THEN 150
127 IF P<13 THEN 160
140 GOTO 30
150 POKE -16294,0:POKE -16295,1
155 GOTO 30
160 POKE -16296,0:POKE -16293,1
165 GOTO 30
```

程序4-33中，10至120语句为温度检测部分，与上一节介绍的程序4-31类似。125至165语句为温度控制部分。在125语句和127语句中分别设置了控制温度的上限点和下限点，当温度低于13℃时，程序转到160语句，该语句为软开关语句，使AN_0端输出高电平，AN_1端输出低电平，从而通过驱动电路接通加热系统，并断开制冷系统。当温度逐渐上升至13℃～15℃之间，从程序看到，加热及制冷系统将维持原

状不变。当温度继续升高并超过15℃时，由于125语句中设置的条件已满足，程序即转到150语句，该语句也是软开关语句，使AN_0端输出低电平，使AN_1端输出高电平，从而通过功率驱动电路断开加热系统，并接通制冷系统，使温度下降。30语句是每次温度采样后的软件延时语句，延时时长可以根据实际需要来选定。当温度控制的精度要求较高时，可减少循环次数，当温度控制的要求较低时，可适当增加循环次数，这样可相对减少继电器的动作次数，延长继电器的使用寿命。

第十七节　测 量 电 容

电容的测量要比电阻的测量困难一些，一般需使用万用电桥或专用电容测量仪器，这些仪器的价格都比较贵，并不是所有的实验室都能配备的，尤其是一些普通学校。假如我们没有上述仪器，能否用APPLE-Ⅱ微型机来测量电容呢？回答是肯定的。本节将介绍电容测量的原理、接口电路制作以及测量程序。

一、电容测量原理及自制接口电路

这里运用的电容测量原理，类似于前面介绍的电阻值测量，即利用R和C的充放电特性。可是，由于微型机的主机电路上电容C已经固定为0.022μF，所以机内的555时基电路组成的特殊A/D接口电路只能用来测量电阻，而不便用来测量电容。但是我们可以从中得到启发，用自制的A/D接口电路来完成机内电路的相同功能，这样不仅可以更灵活地选换器件，而且还可以增长接口电路的实践知识。

自制接口电路图如图4-51所示。这里我们选用了前面已经介绍过的555时基电路，并在原放置被测电阻的位置上接一固定电阻，所以其电阻值是一个已知量；将被测电容接至原固定电容的位置上，即被测电容C的一端接地，另一端同固定电阻R及555时基电路的6、7脚相接。这样R和C的充放电变化通过6、7脚输入555时基电路，时基电路的输出端第3脚电位取决于6、7脚的电位，由于第3脚接至游戏I/O接口的第2脚，即PB_0输入端上，由此3脚输出电位变化的时间就可以由微型机来统计了。

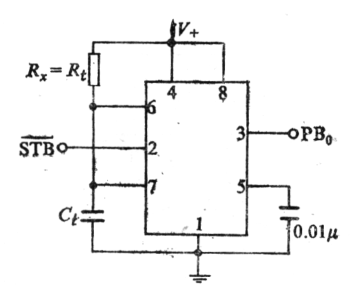


图 4-51 自制接口电路

这里我们还应该了解游戏I/O接口的第5脚功能，通常把该脚称作$\overline{STB}$信号端。

$\overline{STB}$信号是一个实用选通信号，它在程序控制下动作，其用途由用户自己决定，每当访问地址范围＄C040—＄C04F时，信号$\overline{STB}$保持低电平的持续时间为489毫微秒。每访问一次可以得到一个负向脉冲。这里，我们把它接在555时基电路的第2脚上，当程序访问该地址时，负脉冲出现，使555电路内部所含的放电管导通，使7脚电位在瞬间变成0V（此时，3脚输出高电平，微型机开始计数），其效果相当于在外部用导线将电容两端短路后再立即拿开导线。电源电压通过电阻R对电容C充电，当6、7脚的电位达到三分之二电源电压时，3脚即输出低电平，此时微型机计数结束。因为电阻R为一已知固定量，所以微型机的计数将与被测电容量的大小成正比。

在制作接口电路时，先将555时基电路插入电路实验板，并按照图4-51仔细接上元器件和短路线。由于此接口电路所耗电流不大，可以使用主机电源，被测量的电容可以插在实验板上，也可以用夹子线接至实验板。

5脚负脉冲
Y寄存器清零
读3端状态
$PB_0=0?$ Y N
Y=Y+1
溢出吗? N Y
Y−1⇒Y
结果送$08

图 4-52 测时子程序的程序框图

二、测时子程序简介

我们知道,BASIC语言虽具有简单易学，使用方便等显著优点，但在某些实时控制的场合，由于其执行速度太慢而不能使用。本节介绍的电容测量方法，对测时子程序有速度要求，因此不能用BASIC语言编程，可用汇编语言来编写。

测时子程序的程序框图如图4-52所示。其程序清单如下：

程序4-34

```
0300-    AD 40 C0      LDA    $C040
0303-    A0 00         LDY    #$00
0305-    AD 61 C0      LDA    $C061
0308-    10 04         BPL    $030E
030A-    C8            INY
030B-    D0 F8         BNE    $0305
030D-    88            DEY
030E-    84 08         STY    $08
0310-    60            RTS
```

考虑到一部分读者没有学过汇编语言，下面我们将汇编语言同大家已学过的BASIC语言加以对照。

从程序4-34中可以看到，在用汇编语言编写的程序中，所有的数字都用十六进制，而在 BASIC 程序中，数字都 用十进制。

程序中的第一个语句LDA C040是程序访问地址$ C040相当于 BASIC 程序中的PEEK（-16320）语句，使游戏I/O接口的第 5 脚输出一个$\overline{STB}$ 负向脉冲。这里用Y寄存器作为测时计数器，LDY #00 是将Y 计数器清零，这相当于BASIC程序中的LET Y=0。LDA C061 语句中的C061 是游戏I/O的PB_0端的地址，该语句是将555电路的 3 端状态通过PB_0端读入计算机，相当于 BASIC 程序中的 PEEK（-16286）。BPL 030E为分支语句，用来判断PB_0端的状态是否为0。当N标志位为 1 时，执行INY语句；当N标志位为0时，则转到地址030E，该语句STY $ 08 是将测时结果送到地址$ 08存放起来。INY语句相当于BASIC 程序中的 LET Y=Y+1。它后面紧跟的语句BNE 0305 也是分支语句，观其结果是否溢出，即是否超过 8 位二进制的最大数 255 。若没有溢出，则转到地址 0305；若溢出，则执行DEY语句。该语句相当于BASIC程序中的 LET Y=Y-1，所以，没有溢出的测试结果及表示溢出的数255都会在显示器上显示出来。

三、在BASIC程序中直接调用汇编子程序

怎样把程序 4-34 编入用 BASIC 语言编写的程序中去？让我们再仔细分析一下这一用汇编语言编写的子程序。

在该程序清单的右边是符号指令，它很清楚地反映该条指令的功能，左边是同符号指令相对应的机器码及其所在地

址。可以这样认为，从本质上看，不管哪一种语言编写的程序，实际上最终都是一些地址及其对应的内容。前面已经介绍过，APPLESOFT语言中可以使用专用指令POKE，将机器码逐一放入所对应的地址。例如，程序中地址0300相对应的内容为机器码AD，先将这两个十六进制数化成十进制数，它们分别为768和173，然后用POKE 768，173指令把内容173放入地址为768的内存中。所以根据程序中的地址和对应的机器码，就可以把这一子程序用POKE指令放入内存。我们再看几个对照例子，如地址0301的内容为40，对应于POKE 769,64，地址0302的内容为C0，对应于POKE 770，192等。为了使初学者能了解得更清楚些，下面给出了把子程序放进内存的全部POKE指令：

```
POKE768,173:POKE769,64:POKE770,192:
POKE771,160:POKE772,0:POKE773,173:
POKE774,97:POKE775,192:POKE776,16:
POKE777,4:POKE778,200:POKE779,208:
POKE780,248:POKE781,136:POKE782,132:
POKE783,8:POKE784,96
```

因为这里的地址是个连续增加的正整数，所以可以用循环语句简化这个程序。简化后的程序清单如下：

程序4-35

```
20  FOR I=768 TO 784: READ BYTE: POKE  I, BYTE:
    NEXT I
100 DATA 173,64,192,160,0,173,94,192,16, 4, 200, 208,
    248,136,132,8,96
```

程序放入内存后怎样调用它呢？APPLE SOFT BASIC中提供了CALL这个专用指令，在CALL后面跟上地址，例如CALL 768，BASIC程序就调用了首地址为768的这段子

程序，一直到该子程序执行完毕后才回到BASIC程序。子程序执行的结果，可用PEEK语句从结果存放的地址中取出。

四、程序简介

由于这里介绍的电容测量原理同电阻测量原理相同，所以它的计算式也是一样的，$C=(M*10.8)11.1*R$，式中M为实测的累计次数，电阻R的值可以根据实测电容的范围灵活选择，下面程序中所采用的电阻实测到10.04kΩ，程序清单如下：

程序4-36

```
10  CALL  -936
20  FOR I=768 TO 784: READ  BYTE:  POKE  I,BYTE:
    NEXT I
30  S=0:V=0:DIM R(19)
60  FOR I=0 TO 19: CALL 768:R(I)=PEEK(8)
70  S=S+R(I):NEXT I
80  M=S/20:C=(V*10.8)/(1.1*10040)
85  PRINT  "M=";M;"     C=";C
90  S=0:GOTO 20
100 DATA 173,64,192,160,0,173,97,192,16, 4, 200, 208,
    248,136,132,8,96
```

在程序4-36中，10语句为屏幕清除语句，20语句将汇编语言子程序放入相应的内存，30语句将S、M清0，60语句中用CALL语句调用子程序，并用PEEK(8)语句将子程序执行结果从地址＄08中取出，80语句将测试数据转换成电容量，再由85语句打印输出。100语句提供了地址768～784相对应的内容。

五、自制A/D接口的灵活运用

前面已介绍了，在使用主机的简易A/D接口测量电阻值时，被测电阻值的范围有一定的局限性，其原因是，被测电阻值是由算式$R=(M\times10.8)/(1.1\times0.022)$给出的，而$M$的最大值为255，可以算出$R$的最大值约为113.8kΩ。如被测电阻的阻值大于此值，可将0.022μF的电容换小一些，如果把它改换成0.01μF，电阻的最大值可以测到250kΩ，在自制的A/D接口电路上更换电容十分方便。相反地，当增大电容C时，能使电阻值的测量范围向低阻方向转移，这是因为：当M为2时，微型机已接近不能分辨，可以算出M为2时R值约为892Ω。若把电容C增大到0.57μF，当M为3时，电阻为51Ω。

掌握了这一特点，我们就可以根据被测电阻的大致范围，来选择合适的电容容量，以得到最佳的测量效果。

六、十六进制数—十进制数的转换程序

使用POKE语句可以直接在BASIC程序中调用汇编语言编写的子程序，这一方法给初学者运用一些汇编语言子程序提供了很大的方便。由于汇编语言中的地址和内容都是用十六进制数表示的，而BASIC程序中的数都是用十进制数来表示，这里就存在着两者的换算问题。假如只有几个数字，用人工换算并不困难，当数字较多时，其工作量就不小了，好在有微型机，让我们进一步发挥它的特长，用它来保质保量地完成这一枯燥的换算任务吧。

下面是将十六进制数自动转换成相应十进制数的程序：

程序4-37

```
10  REM THIS IS HEX-DECIMAL CONVERT PROGRAM
30  PRINT "PLEASE INPUT YOUR HEX-DECIMAL-FIG-
    URE";
32  INPUT S$
35  L=LEN(S$)
37  GOSUB 300
40  IF L=1 THEN D=J1:GOTO 70
45  IF L=2 THEN D=J1*16+J2:GOTO 70
50  IF L=3 THEN D=J1*16*16+J2*16+J3:GOTO 70
60  IF L=4 THEN D=J1*16*16*16+J2*16*16+J3*
    16+J4:GOTO 70
70  PRINT "ORIGINAL INPUT=";S$
100 PRINT "DECIMAL VALUE=";D
190 PRINT "ANY MORE (TO END PROG RAM TO
    ENTER 0 ELSE 1)";
210 INPUT A
230 IF A=0 THEN 270
250 GOTO 30
270 END
300 J1$=LEFT$(S$,1):I$=J1$:GOSUB 400:J1=X
305 IF L=1 THEN J2=0:J3=0:J4=0:RETURN
310 J2$=MID$(S$,2,1):I$=J2$:GOSUB 400:J2=X
315 IF L=2 THEN J3=0:J4=0:RETURN
320 J3$=MID$(S$,3,1):I$=J3$:GOSUB 400:J3=X
325 IF L=3 THEN J4 =0:RETURN
330 J4$=RIGHT$(S$,1):I$=J4$:GOSUB 400:J4=X
340 RETURN
400 REM STATMENT 110-180 CAN BE USED TO CONV-
    ERT
```

```
402 REM SINGLE HEX-FIGURE INTO DECIMAL
405 IF I$<"A" THEN X=VAL(I$):RETURN
410 IF I$="A" THEN X=10:RETURN
420 IF I$="B" THEN X=11:RETURN
430 IF I$="C" THEN X=12:RETURN
440 IF I$="D" THEN X=13:RETURN
450 IF I$="E" THEN X=14:RETURN
460 IF I$="F" THEN X=15:RETURN
470 RETURN
```

程序4-37中的符号＄S表示输入的十六进制数，符号D表示转换成的十进制数。程序运行后，只要从键盘上输入十六进制数后，微型机即给出它的十进制数，若还要输入数时，请按“1”键，若不需要输入时，请按“0”键。

第十八节　智力竞赛装置

不少智力竞赛采用抢答方式，答对了加分，答错了扣分。采用抢答方式必须装备抢答机。

抢答机一般用晶体管或集成电路制作。抢答机的输入端连至多个按钮开关，一旦某个开关被按下后，其它开关就不再起作用了，显示器上立即显示出最先按下开关的抢答者的编号。也就是说，抢答机具有选择和显示两种功能。无论是选择电路还是显示电路都比较复杂，制作起来很费事。

根据前面所学的微型机知识，我们不难发现，用APPLE-Ⅱ微型机的游戏I/O接口作为输入端来设计这样的装置，将会使问题变得十分简单，甚至不需要另外制作接口电路。这里充分利用了该机的潜在功能，用游戏I/O接口的开关量

输入端作为按钮开关的输入端，并发挥微型机的软件优势，用循环语句完成开关状态的巡回检测，用条件语句来代替选择电路，并用低分辨率和高分辨率作图方式造字，其终端显示的优点尤为特出。

一、三路输入的抢答装置

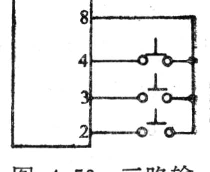


图 4-53 三路输入的抢答装置

游戏I/O接口的开关量输入端共有三个，PB_0、PB_1和PB_2端。先让我们设计一个有三个参赛者的抢答装置。如图4-53所示，将三个按钮开关用导线接至游戏I/O接口。三路输入的控制程序框图如图4-54所示。其程序清单见程序4-38：

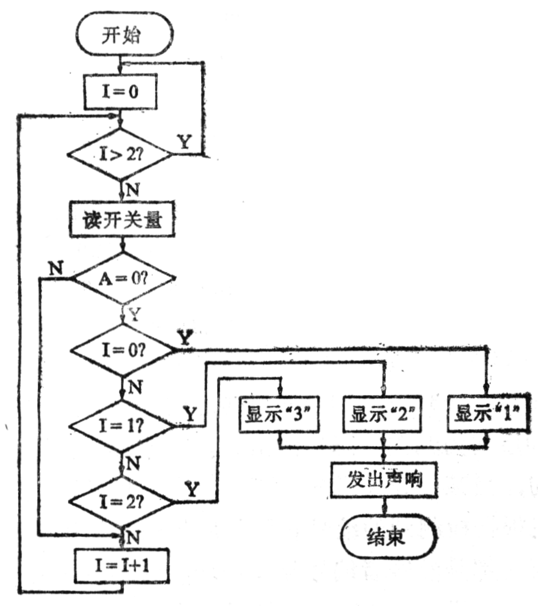


图 4-54 三路输入的控制程序框图

程序4-38

```
5   HGR
10  FOR I=0 TO 2
20  A=PEEK (-16287+I)
30  IF A<>0 THEN 65
40  IF I=0 THEN 70
50  IF I=1 THEN 80
60  IF I=2 THEN 90
65  NEXT I
68  GOTO 10
70  FOR I=1 TO 100:HPLOT 120,40,+I TO 140, 40+I:
    NEXT
75  GOTO 100
80  FOR I=1 TO 20: HPLOT 100, 40+I TO 160, 40+I:
    HPLOT 140, 60+I TO 160, 60+I:HPLOT 100, 80+I
    TO 160, 80+I: HPLOT 100, 100+I TO 120, 100+I:
    HPLOT 100,120+I TO 160,120+I:NEXT
85  GOTO 100
90  FOR I=1 TO 20: HPLOT 100, 40+I TO 160, 40+I:
    HPLOT 140, 60+I TO 160, 60+I: HPLOT 100, 80+I
    TO 160, 80+I: HPLOT 140, 100+I TO 160, 100+I:
    HPLOT 100,120+I TO 160,120+I:NEXT
100 FOR I=1 TO 500 : P=PEEK (-16336) : NEXT
```

在程序4-38中，10至65语句之间的循环语句用作巡回检测。20语句为开关状态检测语句。在循环体内还设有条件判断语句，当某一开关最先按下时，有关信号即被读入机内，微型机根据检测到的编号转到相应的输出显示语句。70、80和90语句都是参赛者编号显示语句，它们利用APPLE-Ⅱ微型机的高分辨作图功能，分别显示出1、2、和3字，即使

坐在后面的观众，也能看清楚显示屏上抢答者的编号。在显示编号的同时，还用100语句使主机发出声响。

二、十二路输入的抢答装置

有时参赛者较多，三路输入装置不够使用。这里介绍一种仍然使用游戏I/O接口，但其输入可扩展到十二路抢答装置。下面先介绍十二个按钮开关与微型机之间的连结。

如图4-55所示，从游戏I/O接口的开关量输入端 PB_0～PB_2 端引出三根接线，从它的开关量输出端 AN_0～AN_3 端上分别引出四根接线，并按图接上十二个按钮开关。不难看出，这些开关之间都是独立的，这种连结就是常用的矩阵译码方法。

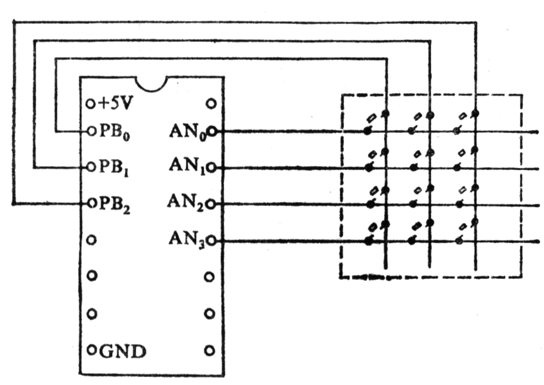


图 4-55　十二个按钮开关与微型机之间的连接

三路输入的检测原理容易理解。扩展至十二路以后，再来判断哪一个开关先按下就要复杂一些。这里我们采用了逐路检测的方法，先用软开关语句将 AN_0 端的输出设置为低电平，同时将另外三端 AN_1、AN_2、AN_3 端设置为高电平，然后运用检查三路输入类似的方法，逐个地检查 PB_0～PB_2 同

AN_0之间分别连结的开关。若检测到某一开关已被按下，显示器即输出该开关的位置号；若没有开关按下，则将AN_1端设置为低电平，其余三端设为高电平，再继续巡找，一直到某一开关按下后，这个过程才结束。

十二路输入的控制程序框图如图4-56所示。其程序清单见程序4-39：

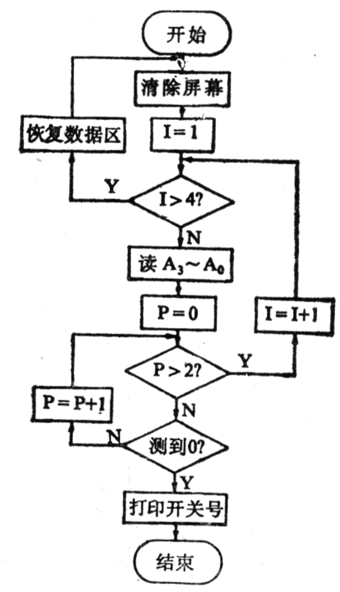


图 4-56 十二路输入的控制程序框图

程序4-89

```
5    CALL  -936
10   FOR I=1 TO 4
20   READ A3,A2,A1,A0
30   POKE(-16296+A0),0
40   POKE(-16294+A1),0
50   POKE(-16292+A2),0
```

```
60  POKE(-16290+A3),0
70  FOR P=0 TO 2
80  IF PEEK(-16287+P)=0 THEN 200
90  NEXT P
170 NEXT I
180 RESTORE
190 GOTO 5
200 PRINT"          S";I;P;"     IS CL OSE"
210 DATA 0,1,1,1,1,0,1,1,1,1,0,1,1,1,1,0
220 END
]RUN
    S11  IS CLOSE
]RUN
    S21  IS CLOSE
```

程序4-39中的5语句为屏幕清除语句，这里通过改变软开关语句中变量A_0～A_3的数据，来改变地址，达到改变AN_0～AN_3端输出状态的目的。为了使AN_0～AN_3端状态逐一改变，在210语句中共放置了四组数据。以第一组数0、1、1、1为例，程序读入后，使AN_0端输出低电平，而使其它三端输出高电平，70～90语句中用内循环来检测PB_0～PB_2端的开关输入状态。一旦有开关按下时，即转向200语句，该语句为打印输出语句，将P和I的号（即开关所在的序号）输出。若没有开关按下，程序通过外循环将DATA语句中的第二组数1、0、1、1读出，并进入内循环继续巡找，这个过程一直进行到某一开关按下为止。

三、APPLE-Ⅱ显示屏作智力竞赛记分牌

要举办智力竞赛没有电动记分显示板怎么办？在

APPLE-Ⅱ微型机上运行程序4-40，就能将您的计算机屏幕变成一个有多种显示功能的计分板，该显示装置有如下一些主要功能：

1．能用大字显示出各组的得分及每次加分或扣分的数字；

2．能按组号的顺序显示出各组的得分；

3．能自动按各组得分的多少排出名次；

4．能用大字显示出（按名次）各组的得分和名次。

利用该机的低分辨率图形功能组成数字，在12吋的屏幕上显示出一至三个从0～990的数字，在容纳五、六百人的礼堂内，即使在后排也能看得很清楚。组号和答题应加减的分数用小字分别显示在下方和左下角。

APPLE-Ⅱ显示屏作记分牌的程序清单如下：

程序4-40

```
90   HOME:VTAB 10:HTAB 5:FLASH:PRINT "THE GRO-
     UPS FOR THE GAME";:NORMAL:PRINT"(1-9)?";
100  DIM A(9)
110  GET AA$:B=ASC (AA$) -48:IF B>9 OR B<=
     0 THEN 110
120  PRINT B:PRINT:PRINT:PRINT:PRINT:HTAB (5) :
     INVERSE:PRINT "THE START MARKS";
125  NORMAL:PRINT "(100-900)?";
130  GET AA$:C=ASC(AA$)-48:IF C>9 OR C<=
     0 THEN 130
135  PRINT C*100:FOR I=1 TO 500:NEXT
140  HOME:VTAB(5)
150  FOR I=1 TO B:HTAB(8):PRINT "+++"; I; "+
     ++",C*100:PRINT:A(I)=C*100:NEXT
```

```
170  GET AA$:IF AA$<>"$" THEN 140
180  HOME:VTAB 24: HTAB 16: PRINT "+++ ++
     +";:HTAB 20: GR:COLOR=15
210  GET AA$: IF AA$= "$" THEN TEXT: HOME:
     VTAB 5:FOR I=1 TOB:HTAB 8:PRINT "+++";
     I;"+++",A(I):PRINT :NEXT:GOTO 2000
220  D=ASC (AA$)-48:PRINT D:IF D>B OR D<=0
     THEN 180
240  IF A(D)<0 THEN PRINT A(D):GOTO 4000
243  IF A(D)>900 THEN PRINT A(D):GOTO 3090
245  E=INT (A(D)/100):IF E=0 THEN 1000
250  ON E GOTO 3010,3020,3030,3040,3050,3060, 3070,
     3080,3090
1000 IF A (D) >990 THEN 3190
1005 F=INT ((A(D)-E*100)/10:IF F=0 THEN 3200
1010 ON F GOTO 3110,3120,3130,3140,3150,3160,3170,
     3180, 3190
1500 FOR I=1 TO B:M(I)=A(I):N(I)=I:NEXT:FOR I=
     1 TO B-1:FOR J=I+1 TO B:IF M(I)<M(J) THEN
     O=M(J):M(J)=M(I):M(I)=O:O=N (I) :N (I) =N
     (J):N(J)=O
1520 NEXT J, I:TEXT:HOME:VTAB 5:FOR I=1 TO B:
     HTAB 8:PRINT "+++";N(I);"+++",M(I):PRI-
     NT:NEXT:GET AA$:IF AA$<>"$" THEN 180
1560 ××=2:R=1
1570 HOME:GR:COLOR=15:VTAB 24:PRINT R; :D=N
     (R):HTAB 16:PRINT"+++"; D; "+++" :GOTO
     240
1580 R=R+1:IF R<=B THEN 1570
```

```
1590  TEXT:HOME:VTAB 10:HTAB 6:FLASH
1600  PRINT "BISAI JIXUJINXINGMA?";
1610  NORMAL:PRINT"(Y/N)":GET  AA$:IF  AA$="Y"
      THEN  ××=0:GOTO 180
1620  IF AA$="N" THEN TEXT:HOME:END
1630  GOTO 1590
2000  GET AA$: IF AA$="$" THEN 1500
2010  GOTO 180
3010  VLIN 10,30 AT 7:GOTO 1000
3020  VLIN 12,11 AT 1: HLIN 2,10 AT 10:PLOT 11,11:
      VLIN 12,18 AT 12:FOR I=11 TO 2 STEP-1:PLOT
      I,31-I:NEXT I:HLIN 1,12 AT 30
3025  GOTO 1000
3030  VLIN 12,11 AT 1:HLIN 2,10 AT  10:PLOT  11,11:
      VLIN 12,18 AT 12:HLIN 3,11 AT 19:PLOT 11,20:
      VLIN 21,28 AT 12:PLOT 11,29:HLIN 10,2 AT 30:
      VLIN 29,28 AT 1
3035  GOTO 1000
3040  FOR I=10 TO 19:PLOT 20-I,I:NEXT I:VLIN 20,
      25 AT 1:HLIN 1,12 AT 26:VLIN 10,30 AT 11
3045  GOTO 1000
3050  HLIN 1,12 AT 10:HLIN 1,10 AT 18:HLIN 2,10 AT
      30:VLIN 11,17 AT 1:VLIN 20,28 AT 12:PLOT 11,
      19:PLOT 11,29:PLOT 1,29
3055  GOTO 1000
3060  HLIN 3,11 AT 10:HLIN 3,10 AT 18:HLIN 3,10
      AT 30:VLIN 12, 28 AT 1:VLIN 28, 20 AT 12:
      PLOT 2,11:PLOT 12,11: PLOT 2,19:PLOT 11,19:
      PLOT 2,29:PLOT 11,29
```

```
3065  GOTO 1000
3070  HLIN 1,12 AT 10:VLIN 17,30 AT 7: PLOT 1,11:
      FOR I=12 TO 15:PLOT 23-I,I: NEXT:GOTO 1000
3080  HLIN 3,10 AT 10:HLIN 3,10 AT 19:HLIN 3,10 AT
      30:VLIN 13,17 AT 1:VLIN 13,17 AT 12:VLIN 21,
      28 AT 1:VLIN 21,28 AT 12:PLOT 2,11:PLOT 11,
      11:PLOT 2,18:PLOT 11,18:PLOT 2,20:PLOT 11,
      20:PLOT 2,29:PLOT 11,29:GOTO 1000
3090  HLIN 3,10 AT 10:HLIN 3,10 AT 19:HLIN 2,10 AT
      30: VLIN 12,17 AT 1:VLIN 12,28 AT 12:PLOT 2,
      11:PLOT 11,11:PLOT 2,18:PLOT 11,18:PLOT 1,29:
      PLOT 11,29
3095  GOTO 1000
3110  VLIN 10,30 AT 20:GOTO 4000
3120  VLIN 12,11 AT 15:HLIN 16,24 AT 10:PLOT 25,11:
      VLIN 12,18 AT 26:FOR I=25 TO 16 STEP-1:
      PLOT I,45-I:NEXT I:HLIN 15,26 AT 30
3125  GOTO 4000
3130  VLIN 12,11 AT 15:HLIN 16,24 AT 10:PLOT 25,
      11:VLIN 12, 18 AT 26: HLIN 17, 25 AT 19:PLOT
      25,20: VLIN 21,28 AT 26: PLOT 25,29:HLIN 24,
      16 AT 30:VLIN 29,28 AT 15
3135  GOTO 4000
3140  FOR I=10 TO 19:PLOT 34-I,I:NEXT I: VLIN 20,
      25 AT 15:HLIN 15,26 AT 26:VLIN 10,30 AT 25
3145  GOTO 4000
3150  HLIN 15,26 AT 10:HLIN 15,24 AT 18:HLIN 16,24
      AT 30:VLIN 11,17 AT 15:VLIN 20,28 AT 26:PLOT
      25,19:PLOT 25,29:PLOT 15,29
```

```
3155  GOTO 4000
3160  HLIN 17,25 AT 10:HLIN 17,24 AT 18:HLIN 17,24
      AT 30:VLIN 12, 28 AT 15:VLIN 28, 20 AT 26:
      PLOT 16,11:PLOT 26,11:PLOT 16,19:PLOT 25,19:
      PLOT 16,29:PLOT 25,29
3165  GOTO 4000
3170  HLIN 15,26 AT 10:VLIN 17,30 AT 21:PLOT 15,11:
      FOR I=12 TO 15:PLOT 37-I,I:NEXT
3175  GOTO 4000
3180  HLIN 17,24 AT 10:HLIN 17,24 AT 19:HLIN 17,24
      AT 30:VLIN 13,17 AT 15:VLIN 13,17 AT 26: VLIN
      21,28 AT 15:VLIN 21,28 AT 26:PLOT 16,11: PLOT
      25, 11: PLOT 16, 18: PLOT 25, 18:PLOT 16, 20:
      PLOT 25,20:PLOT 16,29:PLOT 25,29
3185  GOTO 4000
3190  HLIN 17,24 AT 10:HLIN 17,24 AT 19:HLIN 16,24
      AT 30:VLIN 12,17 AT 15:VLIN 12,28 AT 26:PLOT
      16,11:PLOT 25,11:PLOT 16,18:PLOT 25,18:PLOT
      15,29: PLOT 25,29
3195  GOTO 4000
3200  HLIN 17,23 AT 10:HLIN 17,23 AT 30:VLIN 12,28
      AT 15:VLIN 12,28 AT 25:PLOT 16,11:PLOT 24,
      11:PLOT 16,29:PLOT 24,29
4000  HLIN 31,37 AT 10:HLIN 31,37 AT 30:VLIN 12,28
      AT 29:VLIN 12,28 AT 39:PLOT 30,11:PLOT 38,
      11:PLOT 30,29:PLOT 38,29
4010  GET AA$:IF XX=1 THEN XX=0: GOTO 180
4015  IF XX=2 THEN 1580
4020  IF AA$="-" THEN 4100
```

```
4025  IF ASC(AA$)<48 OR ASC(AA$)>57 THEN  4010
4030  H=(ASC(AA$)-48)*10:A(D)=A(D)  +H:XX=1:
      PRINT"+";H: TEXT:GR:COLOR=15:GOTO  240
4100  GET AA$:IF ASC (AA$)-48>9  OR ASC(AA$)
      -48<0 THEN 4100
4120  H=(ASC (AA$)-48)*10:A (D)=A(D)  -H:XX=
      1:PRINT"-";H:GR: COLOR=15:GOTO 240
```

使用程序4-40不必担心按错键，因为按错了计算机是不会理会的，只需重新键入正确的字符。

开机启动程序后，屏幕上显示出“THE GROUPS FOR THE GAME？”询问参赛的组数。参赛组数可在1～9组之间选择，按1～9的数字键。接着屏幕又显示出“THE START MA？”询问本次竞赛的基本分数，范围可在100～900分之间选择，用数字键1代表100分，2代表200分……、9代表900分。给出基本分数后，马上在屏幕上显示出本次比赛各组的组号及其基本分数。

此时只要按下“$”键，便可清屏并在屏幕下方显示出“+ + +　+ + +”等待输入组号。再在数字键上按下相应的组号，屏幕上即用大字显示出该组的基本分。在答题之后，根据竞赛主持人所给的或所扣的分数按下数字键，若扣分则先按“—”键然后再按数字，此时屏幕上的大字即改为加分或扣分后的分数。再按下任意一键，屏幕上又开始显示“+ + +　+ + +”，询问需显示哪一组的分数。如此周而复始，把每次的得分累加起来。

在竞赛中还可随时查询参赛各组的得分情况，只要按下“$”键，就会把参赛各组的组号和所得总分列出表来。按任意一键（除“$”键外）即可返回。如按“$”键，可按

得分的多少将各组的名次排出并列表显示。按除“$”以外的任意一键可返回。

如再按“$”键，则按得分的多少为顺序，将名次、组号和实得分数，用小字和大字分别显示出来，每按一次任意键报出一名，名次报完之后，计算机询问比赛是否继续执行？如需加赛则按“Y”键返回主程序，否则按“N”键结束。

程序中的90～180语句为初始程序，210～2010语句为各种功能的服务程序。3010～4000语句为低分辨率作图的组字程序。4030语句为加分，4120语句为扣分。如有彩色监视器，可将程序中的COLOR语句改成您满意的颜色。

第十九节　脉冲信号发生装置

随着现代半导体技术的飞速发展，数字集成电路的成本已大大降低，脉冲数字技术不仅在数字通讯、电子计算机以及数字仪表等许多领域得到广泛的应用，而且通过家用电器、电子玩具等进入了家庭，因此学习一些数字电路的基本知识在现代信息社会中是很有好处的。

脉冲信号发生器是一种研究数字电路所不可缺少的专用仪器。在数字电路的维修中，使用这种仪器可以较快地找出电路的故障点，在有关计算机组成的教学中，使用该仪器和示波器，可使我们能直观地看到电路中的波形变化，从而加深对计算机工作原理的理解。

但是脉冲信号发生器的价格较高，一般学校不容易配置。本节将介绍利用APPLE-Ⅱ微型机制作的脉冲信号发生装置。如同一般的脉冲信号发生器那样，它也能够输出重复

频率可调，脉宽可调的稳定方波。由于计算机所独有的智能优势，通过软件设计还能使该装置在程序的控制下，自动改变脉冲的重复频率，自动调节脉冲宽度，成为新一代的智能信号装置。

一、脉冲波形的主要参数

在介绍脉冲信号发生装置之前，有必要先学一点脉冲波形的基本知识。

理想的矩形脉冲（常称为方波）如图4-57所示。它由前沿、顶部、后沿等组成，其中两个相邻脉冲的前沿之间（或后沿之间）的时间称为一个周期，一般用T表示，改变T值即改变了它的重复频率。从图中还可以看到，T由脉冲的高电平阶段和低电平阶段所组成（上升沿和下降沿时间极短、可忽略）。当低电平为0电平时，该脉冲称为正向脉冲，当高电平为0电平时，该脉冲则称为负向脉冲，这里我们只讨论正向脉冲。

图 4-57 方波

在实际应用中，除了要求频率可以调节外，还要求脉冲的宽度也能调节，以满足不同电路的需要，所以脉宽可调也是脉冲信号源的主要性能。脉冲波形的测量需要使用示波器。

二、脉冲形成原理及程序简介

从以上介绍的脉冲基本知识中可以看到，脉冲的形成有两个主要过程：一是电平的快速变化阶段，即从高电平变为低电平或从低电平变为高电平；另一阶段是电平的持续，即高电平延时和低电平延时。前面我们已经学习过，用APPLE-

Ⅱ微型机的软开关语句可以方便地使游戏I/O接口的四个开关量输出端输出高电平或低电平。以AN_0端为例，在BASIC程序中，用POKE-16295，1来设置高电平，用POKE-16296,0来设置低电平，而在汇编语言编写的程序中用STAC058来设置高电平，用STA C059来设置低电平，电平的持续时间可用软件延时的方法来实现，这个过程可以用图4-58所示的框图来描述。

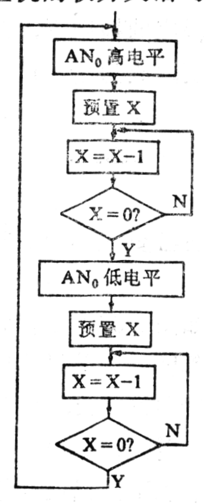


图 4-58 矩形脉冲形成的程序框图

由于BASIC程序的速度慢，这里用汇编语言来编制程序。其程序清单如下：

程序4-41

```
0300—  8D 58 C0    STA   $C058
0303—  A2 7F       LDX   #$7F
0305—  CA          DEX
0306—  D0 FD       BNE   $0305
0308—  8D 59 C0    STA   $C059
030B—  A6 7F       LDX   $7F
030D—  CA          DEX
030E—  D0 FD       BNE   $030D
0310—  20 00 03    JSR   $0300
```

程序4-41中先用STA C058指令使AN_0端输出高电平，将确定其延时时间的数字7F，用LDX指令放入X寄存器中，经过DEX指令减1后由BNE指令加以判断，结果不为0时再继续减1，直至结果为0时退出循环，再执行STA C059

指令，该指令使 AN_0 端输出低电平。和高电平时的情况类同，确定低电平延时时间的数字，也用LDX指令放入X寄存器中，并进入循环状态，直至结果为0时再退出，由JSR转移指令返回到地址0300，使程序反复执行。由此可知，两次存入X寄存器的数值之和决定了脉冲的重复频率，而一次存入X寄存器的数字决定了脉冲的宽度。

为了使初学者更容易理解和实践，我们用 BASIC 程序中的POKE指令写出上述程序。为了便于从键盘上改变输出脉冲的重复频率和脉冲宽度，这里将程序 4-41 稍作修改，增加 INPUT 语句来输入两次存入X寄存器中的数据，并分别用M和N表示它们，但程序的基本结构不变。其程序清单见程序4-42。

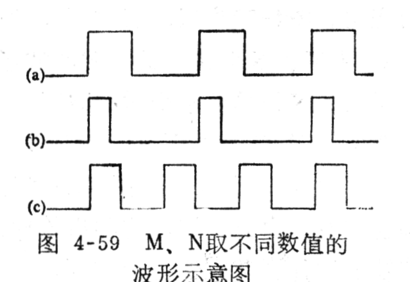


图 4-59　M、N取不同数值的波形示意图

程序4-42

```
10  INPUT M,N
20  POKE 768, 141:POKE 769, 88:POKE 770, 192:POKE
    771,162
30  POKE 772,M
40  POKE 773,202:POKE 774,208:POKE 775,253:POKE
    776,141
50  POKE 777,89:POKE 778,192:POKE 779,162
60  POKE 780,N
70  POKE 781,202:POKE 782, 208:POKE 783, 253:POKE
    784,32:POKE 785,0:POKE 786,3
80  CALL 768
```

三、脉冲波形的测量

在关机状态下，用导线将游戏 I/O接口 的第15脚（AN_0端）和第8脚（接地端）分别接至示波器的探头，然后开机并运行程序4-42。从键盘上输入M和N的值，按回车键后即可在示波器上看到矩形脉冲。为了进一步验证程序的功能，从键盘上输入不同的M、N值，同时用示波器观察脉冲波形的变化。图4-59为M、N取不同数值时的波形示意图，其中（a）为M、N取相同数值时的波形；（b）为减小M，增大N但M＋N的总数不变时的波形；（c）为使M、N都减小后的波形。图（b）与图（a）相比较，其脉宽变窄，但重复频率没有改变。图（c）与图（a）相比较，其脉宽和重复频率都已改变。由此可知，这种用软件来调节输出脉冲的方法比传统的硬件调节方法更为方便。

脉冲的重复频率一般可用数字式频率计直接测量出来，在没有频率计的情况下，也可用脉冲示波器测定脉冲的周期来确定脉冲的频率。

APPLE-Ⅱ微型机产生的方波频率也可以计算出来，因为将X寄存器作为循环计数器时，每次循环时间需11个时钟周期共10.8μs。所以周期T可用（M＋N）×10.8计算出来。如M和N都取127，则周期$T=(127+127)\times10.8=2743\mu s$，频率$f=1/T$，所以该脉冲的重复频率为360Hz。因为X寄存器是八位寄存器，使用上述程序时，X的最大数是255，可以算出它的最低频率约为180Hz；X的最小数只能取1，用相同方法可算出它的最高频率约为46kHz。这样宽的输出频率范围已能基本满足我们的一般需要。

如需调节输出脉冲的幅度，可按图4-60所示，在游戏

I/O接口的第15脚和第8脚之间接入100kΩ电位器，从电位器的中心抽头端输出脉冲。由于微型机产生的脉冲信号频率范围的低端在音频信号范围内，可将该信号输入到收音机的低频放大级，从扬声器中监听音频叫声及其频率的变化。

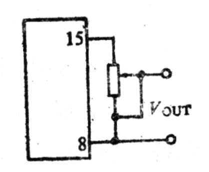


图 4-60 输出脉冲幅度的调节

对汇编语言有兴趣的读者，也可以用APPLE-Ⅱ微型机的小汇编程序输入和运行用汇编语言编写的程序4-41，将得到相同的实验结果。考虑到部分读者对小汇编程序并不熟悉，下面将对其使用方法作有关介绍。

四、小汇编程序

所谓汇编程序，就是将汇编语言程序转换翻译成机器语言的程序。因为汇编语言程序可用记忆符及特定格式，所以无论编写程序和侦错都比机器语言容易得多。在APPLE-Ⅱ微型机的整数BASIC文件中提供了一个小型汇编程序，使得用户可以直接写入汇编语言形式的程序。所以在启动时，需先用INT命令调出整数BASIC，然后方可使用小汇编。

有关操作如下：

1．使机器进入小汇编控制之下：插入带INT BASIC文件的磁盘后，用以下操作即可使APPLE-Ⅱ进入小汇编：

]INT　此时显示提示符>，表示已进入整数BASIC控制之下。

>CALL—151　此时显示提示符*，表示已进入监控程序控制之下。

*F666G　此时显示提示符!，表示已进入小汇编控制之下。

注意：整数BASIC的提示符是“>”

浮点BASIC的提示符是“]”

监控提示符是“*”

小汇编提示符是“！”

2．在规定内存地址送入符号指令：例如，在300单元送入符号指令CLD，则操作如下：

！300:CLD　APPLE-Ⅱ汇编后将符号指令及机器码显示如下：

0300-　D8　CLD

如继续在301单元送入符号指令LDY#$FF时，可在小汇编提示符后面打入一个空格，再键入符号指令：

！　LDY#$FF

0301-　A0　FF　　LDY　#SFF

按以上操作用小汇编命令送入源程序时，小汇编就把每条符号指令及其对应的机器码连同指令所在地址逐条显示出来。

3．从规定的起始地址列出源程序及对应的机器码：例如，要列出刚才打入的首地址为300的那段程序的符号指令及机器代码，操作如下：

！$0300L

4．将一程序送入规定的存储区域后运行此程序：例如，要运行已送入首地址为300单元的程序，操作如下：

！$0300G

程序执行完毕后，如需要可返回监控程序，用监控命令来检查程序运行结果，检查某些内存单元内容和各寄存器内容是否正确等。

5．检查内存单元内容：例如，要检查300单元的内容，

操作如下：

! $0300

6．修改内存单元内容：例如，要修改从300单元开始的连续5个单元内容，使皆为00，操作如下：

! $0300:00 00 00 00 00

最后介绍在APPLE-Ⅱ微型机上，BASIC、监控及小汇编如何互相转入：

1．由BASIC转入监控：

] CALL-151 屏幕显示“*”作为响应

2．由监控进入小汇编：

*F666G 屏幕显示“！”作为响应

3．由小汇编返回监控：

! $FF69G 屏幕显示“*”作为响应

4．由监控返回BASIC：

*3DOG 屏幕显示“]”或“>”作为响应

5．由小汇编返回BASIC：

! $03DOG 屏幕显示“>”作为响应

! INT 屏幕显示“>”作为响应

! FP 回到浮点BASIC，屏幕显示“]”作为响应

无论是在监控或小汇编状态，按 RESET 键都可回到BASIC。对没有使用DOS3.3操作系统的机器，只能通过按 RESET 键才会由监控返回BASIC。

第二十节　直流马达的调速

一、直流马达的调速原理

直流马达广泛应用于需要改变速度的电气设备中，它的转速同所加的电压成正比。即电压越高，其转速越快，电压越低，其转速越慢。尽管实现直流马达调速的方法较多，如可控硅无级调速、功率驱动电路调速等。不同的调速原理所产生的波形也不同，但就平均电压而言，本质上没有多少差别。

用波形发生电路产生方波，并通过调节脉冲宽度来调节马达转速，是一种常用的直流马达调速方法。下面先介绍一下“占空因数”。

占空因数是脉冲波形的一个重要特性。如果用T来表示脉冲的重复周期，t_k来表示正向脉冲的高电平持续时间，那么占空因数$Q=T/t_k$。当T值一定时，t_k的值越小，其占空因数就越大，占空因数也常称作脉冲空度。

图4-61为一正向脉冲示意图，它的底部为零电平，当脉冲的幅度一定时，高电平的持续时间越长，其电压的平均值就越高。假如脉冲的幅度为5V，$t_k=\frac{1}{2}T$，那么它的输出电压平均值为2.5V；当$t_k=\frac{1}{4}T$时，它的电压平均值为1.25

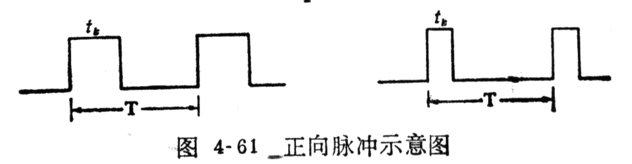


图 4-61 正向脉冲示意图

V，由此可知，占空因数越大，其电压平均值越低。

前面已经讨论过，用APPLE-Ⅱ微型机的游戏I/O接口可以输出脉冲宽度可以调节的方波信号。这种方波是否能直接用来转动马达呢？除了各种直流马达对电源电压的要求有所不同外，这里还有个电路的负载特性问题。电路设计中相当强调负载的影响和匹配，由于马达线圈是较重负载，不能用信号源直接驱动，这里至少需要增加一级功率驱动电路来驱动电机。其接口电路部分如图4-62所示。

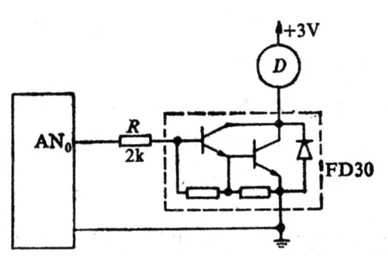


图 4-62 直流马达调速接口电路

这里选用了性能优良的达林顿晶体管作驱动电路，该管的内部由两个放大倍数分别为β_1和β_2的晶体管复合而成，它的总电流放大倍数等于$\beta_1 \times \beta_2$，因此可高达2000倍以上。这样当输出同样电流的情况下，其基极输入电流相对减小，因而对前级电路的影响也减小。同时在该管的内部还有分流电阻和保护二极管，使晶体管的热稳定性和可靠性进一步提高，也减少了电路的外接元件数。

二、程序简介

程序4-43是马达控制程序的清单。

程序4-43

```
5   REM MOTOR SPEED CONTROL
10  INPUT N
20  POKE 768,141:POKE 769, 88: POKE 770, 192:POKE
```

```
        771,162
30  POKE 772,250—N
40  POKE 773,202:POKE 774,208: POKE 775,253: POKE
        776,14:
50  POKE 777,89:POKE 778,192:POKE 779,162
60  POKE 780,N
70  POKE 781,202:POKE 782,208 :POKE 783,253 :POKE
        784,32:POKE 785,0: POKE 786,3
80  CALL 768
```

该程序与前面介绍的信号发生装置程序基本相同，略有不同之处是，将脉冲宽度设为N，重复周期T值设为250，在30语句中用“250—N”来计算脉冲的底部。这样，改变脉宽N时，周期T保持不变。

三、接口电路制作

在本接口电路的制作中，达林顿管是关键器件。这里选用FD30，其耐压可达100V以上，最大集电极电流可达5A。在关机状态下，将达林顿管和电阻插入电路实验板，马达选用1.5伏玩具电机，由于FD30的饱和压降约为1.4V，所以将接口电路的电源电压取3V～3.5V。值得注意的是，马达工作电压虽低，但在高速旋转时的电流可达1安培，所以必须使用机外的直流稳压电源作接口电路电源，切不可使用主机的电源，以防损坏主机。在微型机的相应输出端接上示波器以监视其输出波形。

认真检查所有的连接线，然后开启外接直流稳压电源，此时马达不转。这时可以打开主机，输入并运行程序4-43。当从键盘输入的N值取50时，按回车键后，马达开始慢速转

动，此时，稳压电源的电流表指示在0.1A左右。又取N值为200重新运行时，马达即开始高速运转，稳压电源的电流表指示增加到0.8A～1A，从示波器可清楚地看到不同宽度的脉冲对马达转速所产生的影响。要使马达停止运转，可按RESET键。

用BASIC语言中的软件延时语句，也可以编写输出脉宽可调方波的程序。由于BASIC语言的执行速度慢，所产生的方波频率较低，虽然也可以运用调节占空因数的原理来调节马达的转速，但在慢速运转时，马达的转速明显不均。不过，作为实验，有兴趣者也不妨一试。

用BASIC语言编写的程序清单如下：

程序4-44

```
10  INPUT M,N
15  POKE-16295,1
20  FOR I=1 TO M:NEXT
30  POKE-16296,0
40  FOR I=1 TO N:NEXT
50  GOTO 15
```

第二十一节　步进马达的控制

步进马达适用于控制精度要求较高且所需功率较小的场合，在精密仪器仪表、自动机床等众多领域得到广泛的应用。以前，步进马达一般用脉冲数字电路来控制，现在，不少设计师更喜欢用微型机或微处理机来控制，无疑这将是今后发展的方向。

这里介绍一个利用APPLE-Ⅱ微型机的游戏I/O接口来

控制步进马达的实验，它既能够控制马达的旋转方向，又能控制马达的旋转速度。该系统的自制接口电路简单，控制功能较强。

一、电路简介

为了实现对马达的调速和调向，在游戏I/O接口和马达之间需要自制一个接口电路，图4-63所示即为该部分的电原理图。

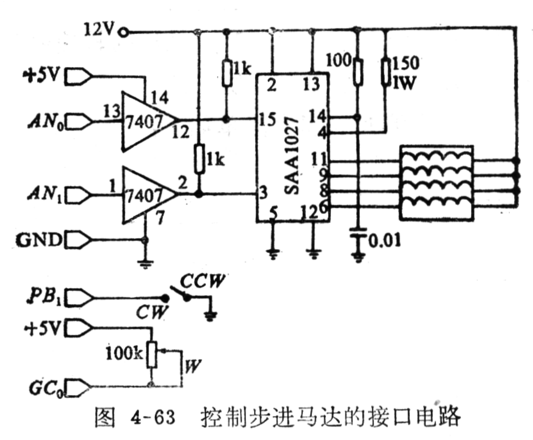


图 4-63 控制步进马达的接口电路

在这一接口电路中，只用了 SAA1027和7407 二块集成电路和几个电阻电容元件。

SAA1027是一种用于控制步进马达的专用集成电路，除了电源输入端外，它还有三个输入端（R、S、T端）和四个输出端（Q_1、Q_2、Q_3、Q_4 端）。R端是它的旋转方向控制输入端，当该端输入为高电平时，马达按逆时针方向旋转；输入为低电平时，马达按顺时针方向旋转。S端是它的输入预置端。当该端为低电平时，它允许输出开关顺序调到预先确定的状态。T端是它的触发输入端，加至该端的脉冲信号控

制马达的转速和位置。Q_1～Q_4输出端分别接至步进马达的四组线圈。

7407是TTL集成电路，这里用作跟随器来提高AN_0和AN_1端的负载能力。

由于所用的12V电源还要供步进马达使用，负载电流达1.5安培，所以必须使用外接直流稳压电源，但外接电源的接地端必须同微型机的地端相接。图4-63中的开关CW为普通开关，用来控制步进马达的转向，W为普通100kΩ的电位器，用来调节步进马达的转速。

二、程序简介

因为步进马达的控制有速度要求，所以程序用汇编语言编写，程序4-45是它的清单。

程序4-45

```
        $107FL
12   107F—     AC 62 C0     LDY   $C062
13   1082—     30 05        BMI   $1089
14   1084—     8D 5A C0     STA   $C05A
15   1087—     10 03        BPL   $108C
16   1089—     8D 58 C0     STA   $C058
17   108C—     AD 58 C0     LDA   $C058
18   108F—     AD 59 C0     LDA   $C059
19   1092—     A2 04        LDX   #$04
20   1094—     AD 70 C0     LDA   $C070
21   1097—     AD 64 C0     LDA   $C064
22   109A—     10 03        BPL   $109F
23   109C—     E8           INX
24   109D—     D0 F8        BNE   $1097
```

```
25  109F—    A0 CB     LDY  #$CB
26  10A1—    88        DEY
27  10A2—    D0 FD     BNE  $10A1
28  10A4—    CA        DEX
29  10A5—    D0 F8     BNE  $109F
30  10A7—    F0 D6     BEQ  $107F
```

程序中的第12～16条指令用来确定马达旋转方向。当PB_1端为高电平时，马达逆时针方向旋转；当PB_1端为低电平时，马达顺时针方向旋转。第17、18条指令先使AN_0端输出低电平，然后输出高电平，使得步进马达在所选择的旋转方向上旋转一步。第19～24条指令用来测量接至GC_0端的电位器阻值情况，累计在X寄存器中的数值用来控制步与步之间的时延。第30条指令使程序重复执行。

按图4-63接好连线，输入并运行程序4-45，然后开启12V外接电源，马达即开始旋转。用CCW/CW开关变换马达的旋转方向，用电位器W调节马达的旋转速度。

第二十二节　简易时钟卡

在使用APPLE-Ⅱ微型机编制的教学和其它应用软件中常常需要用时间控制程序的运行。但由于软件本身不可避免的跳跃性，不对称和键盘输入等待等原因，使应用软件准确地计时变得困难，因此需用外设来协助。本节介绍了简易时钟卡的制作方法及其应用。这块时钟卡成本很低，结构简单、制作容易，但却具有满意的效果。

一、电路简介

简易时钟卡的电原理图如图4-64所示。

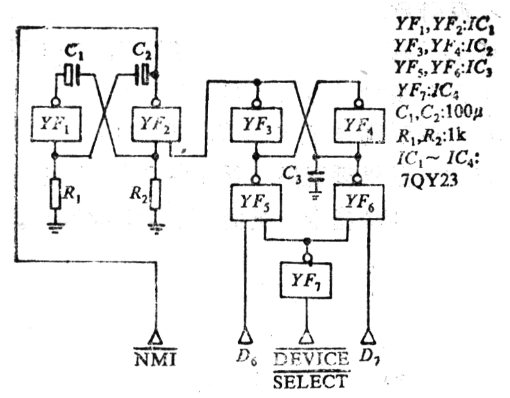


图 4-64 简易时钟卡电路图

在APPLE-Ⅱ微型机槽口的五十条接脚中，有两根中断请求脚，它们分别是$\overline{IRQ}$和$\overline{NMI}$。其中$\overline{IRQ}$脚是可屏蔽中断脚，$\overline{NMI}$是不可屏蔽中断。当$\overline{NMI}$脚出现高电平至低电平的负跳变后，就会被 6502 所识别。由于它是不可屏蔽中断请求，因此6502在执行完正在执行的命令之后，便暂停主程序而进入$\overline{NMI}$中断响应周期。它将程序计数器 PC 值和标志寄存器 P 值送入堆栈保存，然后设定中断禁止标志。最后从 $FFFA和 $FFFB 两单元指示的地址（$3FB）开始运行中断处理程序。

现在以（$3FB）为入口存一个“累加器”程序，在槽口上每间隔一定时间（如 1 秒）触发$\overline{NMI}$，每触发一次，程序使其地址（$314，$315）加 1，从这个地址便可以得到准确的时间。

图4-64中，YF_1和YF_2组成自激多谐振荡器，YF_2输出端接至$\overline{NMI}$端，其周期为1s。为了控制它的开与关，YF_2的一输入端接在由YF_3至YF_6组成的双稳态触发器上，由数据

线 D_7 和D_6控制开、关。$\overline{\text{DEVICE SELECT}}$ 脚是槽口选通脚，此脚低电平则数据线有效。电容器 C_8 的作用是破坏平衡，使通电后振荡器不马上工作。

当6502中断后跳转至＄3FB，但因＄400以后是屏幕显示区不能存放中断处理程序，所以将程序存在＄300。在＄3FB中存JMP＄300指令（机器码：4C 0300）。其程序清单如下：

程序4-46

```
0300—    48         PHA
0301—    8A         TXA
0302—    48         PHA
0303—    98         TYA
0304—    48         PHA
0305—    EE 14 03   INC   $0314
0308—    D0 04      BNE   $030E
030A—    18         CLC
030B—    EE 15 03   INC   $0315
030E—    68         PLA
030F—    A8         TAY
0310—    68         PLA
0311—    AA         TAX
0312—    68         PLA
0313—    40         RTI
0314—    00         BRK
0315—    00         BRK
```

二、制作方法

在制作时钟卡之前有必要进一步熟悉一下7QY23集成电

路，它是一种TTL门电路，其逻辑电路图如图4-65所示。该集成电路内部有两个相互独立的5输入与非门电路所组成，它的主要功能是，当5个输入端都为高电平时，输出为低电平，其中只要有一个输入端为低电平，输出将为高电平。

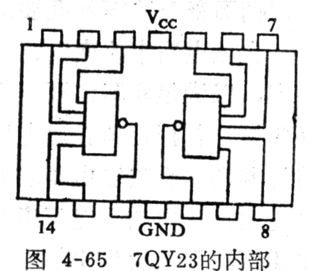
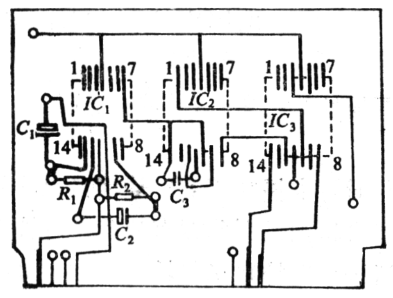
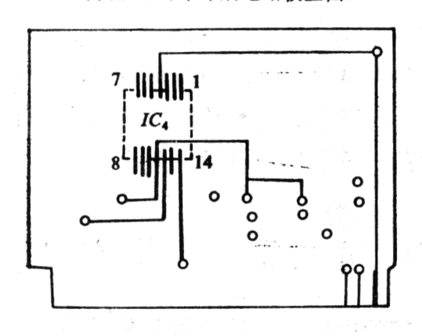


图 4-65　7QY23的内部逻辑电路图

按图4-66制作电路板。这里需用双面敷铜板，图中（a）为它的正面，（b）为它的反面。用双面敷铜板制作电路比



(a)简易时钟卡印刷电路板正面



(b)简易时钟卡印刷电路板反面

图 4-66

单面板的制作要困难一些，需注意两个面上有关线路的一致性，两面都有敷铜的孔要用导线把它们连结起来。

在该电路中改变C_1、C_2、R_1和R_2可改变振荡器的谐振频率，以满足实际需要，但R_1、R_2不要大于20kΩ，否则不易起振。若振荡器在通电后马上开始工作，可将C_3加大，并将控制端接在双稳电路的另一输出端上。若元件合格，且安装无误，该电路一般不用调试即可正常工作。当然简易时钟卡也可以用其它TTL与非门来制作，但电路板需重新设计。

两个控制命令是：1.POKE—16208，128（用于打开时钟卡）；

2.POKE—16208，64（用于关闭时钟卡）。

两个重要地址是：＄314（数字低两位）；＄315（数字高两位）。

关机后将时钟卡插在“3”号槽口上，然后开机，机器自检，如不能自检则是振荡器有故障，可按上述办法排除后再试。正常后进入监控状态并输入程序，最后在＄3FB存入4C 03 00，这时键入POKE—16208，128，时钟卡应工作，＄314与＄315中的数应递增。

这块时钟卡可以在多方面灵活应用，如应用在速算比赛和限制时间的答题中。只要你记住两个命令和两个地址，再发挥你的智慧，就会出现奇妙的效果。

第二十三节 计算机眼睛

本节介绍的电路可使计算机与某种外部设备之间的连接不再需要用硬接线，而是通过光电原理的耦合作用来实现，

这在某些应用中是必需或必要的。例如，计算机要控制一台电机工作一段时间，只需通过控制程序中的几行语句，使显示器的某一区域出现一个或者几个小光点，小光点的出现时间长短由软件控制，这些小光点信号由电路接收后经处理即可控制电机动作。该电路简单可靠，且不受计算机型号的限制，只要有显示屏或数码管即可。

一、电路基本工作原理

该电路主要由光电比较器组成，电路图见图4-67。

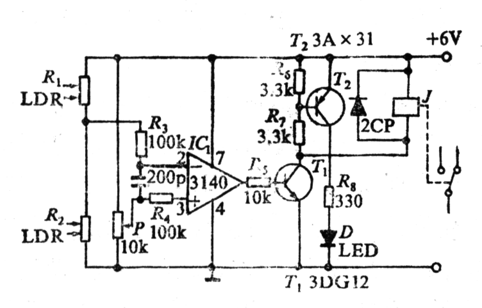


图 4-67　计算机眼睛的电路图

所谓“眼睛”指的是两个光敏电阻（LDR）R_1、R_2。它们连接点的电压通过R_2送到比较器IC_1的反相输入端，IC_1的正相输入端的电位保持在某一基准电压上，该点的电压可用电位器P调整。当反相输入端的电压低于基准电压时，比较器翻转，晶体管T_1导通，继电器得电动作，这时，T_2也导通，LED亮表示继电器得电；当比较器反相输入端的电平高于基准电压时，继电器失电，LED暗。

该电路的工作是由程序控制的，它要求程序控制显示屏，并在上面产生一个或二个光点。如为二个发光点，则其中一个光点的发光强度应保持恒定，另一个发光点的发光强度是

可变的（亮或暗）。如为一个发光点，则另一个LDR应接一恒定光强度的指示灯。

LDR光敏电阻应紧贴在显示屏上固定的发光点位置。两个发光点之间的亮度差决定了两个LDR连接处的电压（约2伏）。

二、电路安装与调试

该电路对元件并无十分严格的要求，除LDR外，可装入一个小盒子内。LDR与电路板间的距离可以远一些，但是要用良好的导线连接，LDR要用护套包好，但要留出受光孔，用胶布贴在显示屏上。

比较器的预置方式也不很严格，仅要求光点变换频率为1Hz。这里，只需简单地调P，以满足继电器能按1Hz频率通断。当频率较高时，例如用于数据转换时，P的预置就较为严格。该电路允许的最高光点变换频率还取决于由R_4和C_1组成的低通滤波器，这里应小于10Hz。P的最佳设置方法如下：先在比较器输入端加一频率为8Hz的方波，然后用一个模拟电压表（直流10V档）接到运算放大器的输出端第6脚上，调节P使电压为电源电压的一半。尽管这时电压表指针会有一些抖动，但不会影响读数。如有一台示波器，则调试效果会更好。

电路中，继电器可选用JTX小型通用继电器，如果其规格是直流6V的，则工作电流为150mA，这时T_1采用3DG12、T_2采用3A×31即可。该电路空载时的电流很小，仅为几个mA。

第五章　微型机使用技巧及有关实用制作

第一节　微型机的多终端显示

在教室较大、学生较多等场合下，用一台显示器学生看不清演示内容，需要增加几台显示器。本节将分别介绍用无线发射和有线扩接来增加微型机的终端显示。这两种方法都不需要改动主机和显示器的电路。

一、使用附加的显示发射器

这里先介绍怎样给LASER310微型机附加一个显示发射器。这个简单的发射装置能让计算机周围10米以内的电视机从天线上接到到清晰的图象。这个装置也适合电视天线插口

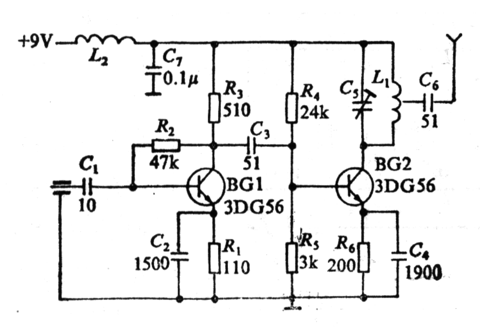


图 5-1　显示发射装置电路图

不配套又不便改装的用户，用直接接收的方式显示计算机输出。

1。电路原理：发射装置的电路图如图5-1所示。整个电路由两个部分组成，左边电路是宽带不调谐高频放大级，计算机输出的射频信号经电容C_1耦合到BG_1三极管的基极，R_1为电流负反馈电阻，R_2为电压负反馈电阻，它们都起稳定直流工作点的作用，并联在R_1上的C_2构成交流通路，以避免交流损失。电路的右边部分为发射输出级，R_4和R_5为偏流分压电阻，R_6为负反馈电阻，用以稳定BG_2三极管的工作点，C_4为旁路电容，C_5和电感L_1组成谐振回路，调谐在计算机的射频频率上（本机为2频道）。两级电路由C_3耦合。L_2、C_7为去耦电路，整机用6F22型9V叠层电池供电，发射天线用一根半米长的普通电线通过C_6接在L_1的抽头处。

2. 元件选择与制作：BG1、BG2三极管应选用高频硅管3DG56（或3DG80，3DG30等），其截止频率f_T应大于700 MHz，放大倍数β应在50～100之间，C_5为瓷介微调电容，C_7为金属化纸质电容，其它均使用高频瓷片电容，电阻均为1/4W金属膜，阻值的要求并不严格。电感L_1可用直径为0.4～0.6mm的镀银裸线或漆包线，在ϕ4钻头上密绕10圈后成为空芯脱胎式，并在中间抽头，L_2用0.07mm的高强度漆包线在1/4W、1MΩ以上的膜电阻上乱绕200圈左右。在安装前，最好能将所有的元件测量一下，

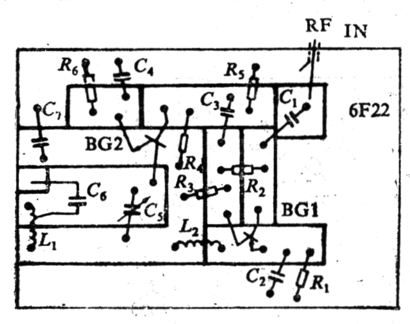


图 5-2 显示发射装置印板接线图

因高频发射装置对元器件的质量要求较高，否则会影响制作效果。

3．安装与调试：为便于制作，这里给出该装置的印刷电路图，如图5-2所示。

高频电路应选用环氧基质板。由于电路比较简单，也可直接用小刀刻成，角上留出四个孔，供固定于塑料小盒内使用。按照电路图5-2安装后，调整 R_2，使 BG1的集电极电流为1.5～2mA，调整R_4，使BG2的集电极电流为2～4mA，将一短段75Ω高频电缆接在输入端头，另一头通过莲花插头接到LASER—310微型机后部的射频输出接口（RF OUT）上，如图5-3所示。

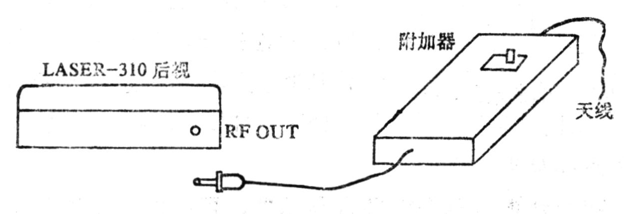


图 5-3　显示发射装置与主机的连接

也可以在输入端接一插座，直接利用LASER—310机的连接线，然后将附加的显示发射器靠近电视机天线，顺序开启电视机和计算机电源，将电视频道开关扳至二频道，用无感螺丝刀缓慢调整C_5，并配合调节电视机的微调旋钮，直至屏幕上出现图象。适当调整L_1的松紧，使图象最为清晰，色彩逼真且无干扰，然后便可拉开距离，进行发射显示试验。

该装置不仅成本低、而且耗电省，整机电流仅为4～5mA。它不仅能够实现LASER-310 机的发射输出，在原理上也适用于COMX、PC-8300或R_1等具有射频输出接口的教

学用微型机。只要按照这些微型机的射频频率调整电路中的谐振回路即可。

二、微型机与多台显示器的有线连接

使用专用显示器和同轴电缆可以将一台主机同若干台显示器连接起来，实现一台主机的多终端显示。

一般显示器的背后都有输入（IN）和输出（OUT）插口，并有一个阻抗变换开关，利用这些插口和开关就能达到多台显示的目的。下面具体介绍连接方法。

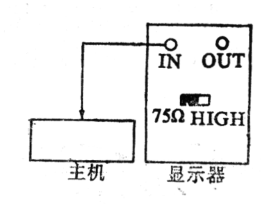


图 5-4 单台显示器与主机的连接

按图 5-4 所示，可将微型机的主机和一台显示器连接起来。也就是用视频信号连接线，将微型机中的视频信号从监视器插口输送到显示器的输入（IN）插口，此时，阻抗变换开关应拨向75 Ω 端。

如一台微型机要配多台显示器，可按图 5-5 所示连接。其中第二台显示器的输入（IN）插口用视频信号线与第一台显示器的输出（OUT）插口联接，第三台显示器的连接方法与第二台相同。如有第四台，连接方法也一样。除最后一台显示器的阻抗变换开关拨向低阻75 Ω 端外，前面几台显示

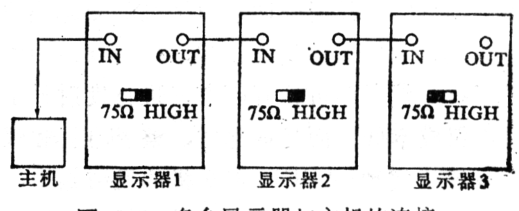


图 5-5 多台显示器与主机的连接

器的阻抗开关都应拨向高阻（HIGH）端。

第二节　两台APPLE-Ⅱ微型机之间的通讯

一般地说，实现多台APPLE微型机之间的通讯是比较复杂且昂贵的。但是，如果在两台APPLE-Ⅱ机之间采用串行通讯接口的方法，再配上相应的通讯软件，那么，实现两机之间的通讯是比较方便的。这里，我们还介绍一种既方便，又经济的方法，即通过普通录音机来实现两机之间通讯的方法。

我们知道，每台APPLE-Ⅱ微型机都有一个录音机输入输出接口，只要我们把输出的微弱信号放大，然后直接接入另一台机器的输入端就能实现信息交换了。这便是利用录音机实现两机通讯的基本原理。这里，录音机的作用是将电信号放大并将它录在磁带上，同时它又起着输出信号的作用。从硬件方面讲，要求录音机必须有线路输入端和耳机输出端，这当然是一般录音机都能具备的。

两机之间相互通讯的输入输出过程是这样的：先准备好两根转录线，其中一根把APPLE机的输出端和录音机的线路输入端（即LINE—IN）连接起来，另一根线把APPLE机的输入端和录音机的耳机输出端（即EAR—OUT）连接起来即可。

在做好上述准备工作后，就可以开始交换信息了。先在要进行信息交换的两台APPLE机上把磁带输入输出的命令打好，注意此时暂不按回车键，然后把录音机上的功能转换开关拨至线路录音功能位置并按下它的RECORD和PLAY

键，这时两台 APPLE 机便可同时按下回车键了

如果交换不成功的话，可以调节一下录音机的音量电位器。如有条件的话，最好能在录音机上安装输出电平监示装置（见本章第七节），这样可给通讯过程带来很大的方便。

假如你使用的是立体声录音机，那么除了用同一声道（左声道或右声道）可以实现两台 APPLE 微型机之间的信息交换外，还可以充分利用它的左右双声道功能，实现一机输出、两机输入或两机输出、两机输入的通讯。具体方法可以参照普通录音机的方法，这里不再详述了。

第三节　微型机的发声扩音方法

LASER-310 微型机具有 SOUND 功能，可以通过编程演奏你喜爱的乐曲。然而，美中不足的是声音微弱。本节介绍的几种扩声方法可以明显地改善其发声效果，凡是利用压电陶瓷做发声器的各种微型机均可参照这些方法适当改造。

一、给微型机加扩音插座

这里我们采用的方法是，将加至压电陶瓷片上的音频电压引出机外，通过低频放大和功率放大后得到较大的输出功率。具体步骤如下：

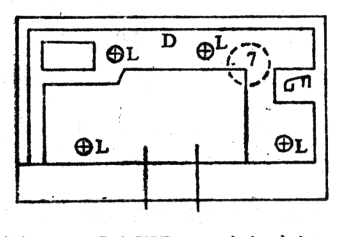


图 5-6　LASER-310主机内部结构示意图

1. 在主机右侧电源开关上方小心地钻一个直径为 5mm 的小孔，用以安装 ϕ3.5mm插座。

2. 将主机底部朝上，用十字螺丝刀把下机壳上的六颗紧固螺钉逆时针方向旋下，

轻轻打开主机。机器可分为上机壳（即键盘板）和下机壳（即主机板）两部分，操作时应注意不要损坏两机壳（即键盘与主机电路板）间的扁平电缆（参看图5-6）。

3．把下机壳内固定印刷板的四个螺钉小心旋下，螺钉位置见图5-6的L处。

4．轻轻掀起整个印刷板（连同上屏蔽板），并在图5-6的D处垫起约70mm高，使印制板成60°的倾斜状态。此时即可看到在下机壳右上角处有一片直径约为30mm的压电陶瓷片（图5-6的T处）。

5．在压电陶瓷片上原焊点处小心而迅速地焊上两条软引线，线的另一端焊接在ϕ3.5插座上。

6．使印制板复位，上紧四只螺钉，圆孔上固定插座。

7．把上机壳与下机壳对缝合好，旋紧六只螺钉，使机器恢复原状。这样就可以用3.5mm的插头直接从插座上取出音频信号了。

需要提醒的是，操作应细心，动作要轻，并且要在环境清洁的室内进行。

现在只要我们用合适的插头，将LASER机与带有拾音器插孔的收音机、扩音机连起来，就可以扩音了。在有条件的学校，如自己安装一块低放电路板作为LASER机扩音专用，将更方便。下面作有关介绍。

二、自制功率放大器

现代半导体技术给我们提供了许多性能优良的集成功率放大器，使早先需要使用很多分立元件才能得到的功率，现在只要用一块体积不大的集成块和很少几个外接元件就可以实现，TDA2003就是其中的一种。

该集成功率放大器仅有五个引出脚，外型与塑封大功率管相仿。它的输出功率大（输出电流I_0高至3. ）、互调失真小、负载阻抗低（最低可至1.6Ω）、外围元件少且安装方便。同时TDA2003还具有完善的保护功能。图5-7是TDA2003的典型应用电路。

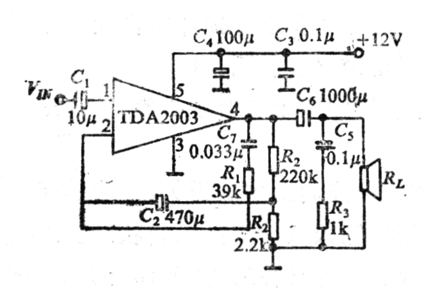


图 5-7 集成功率放大器电路图

图中V_{IN}为它的信号输入端，只要我们把 LASER 机的音频输出接到该端，就可以在扬声器上得到较大的输出功率。TDA2003 还有较宽的电源电压范围，在8V～18V 电压下都能正常工作，和其它集成功放一样，它也需要安装上一定散热面积的散热片。扬声器的功率可根据实际需要选定。

三、用压电陶瓷片作传感器

若不打开 LASER 机的后盖，用压电陶瓷片作传感器也能引出主机的音频信号。买一个直径约27毫米的压电式金属陶瓷片（蜂鸣片），在陶瓷片的内圆（白色部分）及外圆（黄色部分）用烙铁各焊上一根约1.5米的细软导线，将两根导线绞绕成股，在其另一端焊接一个直径为3.5毫米的插头，如图5-8所示。

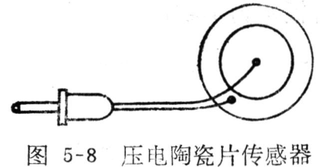


图 5-8 压电陶瓷片传感器

然后取一块约35毫米见方的胶布，把陶瓷片未经焊接的一面紧紧地粘贴在LASER—310机底部的音响处外面，再把

插头插入单声道收录机的AUX插孔（注意不是MIC插孔）。当接通录音机的收音(RAD)钮后，调整音量旋钮，录音机就起音箱作用。微型机奏乐的声音就经录音机放大，且无杂音。

四、用调频机扩音

用一台能收听调频广播的收音机，接通调频（FM）工作状态，放置在LASER-310机旁边的两米范围内，向微型机键入奏乐程序，按下回车键，慢慢旋转收音机的调谐钮，避开正在广播着的电台，注意收听微型机奏乐的声音，同时将天线拔出呈一倾角，并配合调谐加以转动。待从收音机的扬声器中发出微型机奏乐声响、并杂音很小时，再适当调整音量旋钮，一般在45平方米的室内各处均可清晰地听到声音。操作时，收音机距离微型机不可过远，否则效果差，甚至不能扩音。根据试验，在88至108MHz范围内可多处听到微型机的音响，经扩音后发出较为悦耳的电子琴声韵味。

五、用调幅收音机扩音

在条件不具备的情况下，将一台普通的袖珍式半导体中波收音机在接通电源后，紧挨着放在正运行奏乐程序的微型机侧旁（不超过半米范围），慢慢转动收音机的调谐钮，注意避开广播着的电台及杂音大的位置，也可以方便地收听到微型机发出的奏乐声音，再适当调整音量，经过试验，在10kHz～12kHz处效果较佳，不足之处是，有时会遇到杂音干扰，并且在夜晚广播高峰期间几乎听不到微型机的奏乐声音。

六、利用主机的磁带机插孔

主机上的磁带机插孔原用于存取程序。实践经验告诉我

们，如果能在编制软件中注意该插孔的特点，使用合适的语句，也能使该孔输出音频信号，有关程序见程序5-1。

程序5-1

```
10  FOR I= -18738 TO -18688: READ X: POKE I,X:
       NEXT
15  POKE 30862,237:POKE 30863,182:A= USR(0)
20  POKE -18863,179:POKE -18862,182
30  POKE -18762,37
40  POKE 31173,206:POKE 31174,182:POKE 31172,195
100 DATA 229,126,254,58,40,4,35,35,35,35
105 DATA 215,254,158,32,14,209,209,17,0,182
110 DATA 213,35,126,254,58,208,195,125,29,225
115 DATA 201,1,179,0,17,0,182,33,245,43
120 DATA 237,176,1,27,0,33,105,52,237,176
125 DATA 201
```

从主机到磁带机的输出线实际上也是一条现成的音频信号输出通道，它是一根带黑色插头的引线，平时该引线接至磁带机的MIC插口或INPUT 插口，这里将它接至扩音装置。LASER-310机在运行程序 5-1 时（运行后即可NEW掉），凡执行到SOUND语句时，不但电脑能照旧发声，而且LASER—310 机输出线上能输出同样频率峰值为 0.2 伏的音频信号，将信号送至扩音装置，就能发出放大后的声音。

如果没有扩音装置，可用下面的方法试试：将主机与录音机用连接线相连，运行程序 5-1 并清除之，输入反复奏乐的程序 5-2 或者其它奏乐程序，然后按下录音机的录音键，再键入 RUN 命令，这样录音机便可将主机输出的乐曲录下了（请注意，并不是经录音机话筒录下的，这一点不难验证）。该磁带可按普通方法经录音机放音，你就可以欣赏到

先前主机产生的“电子音乐”，其音色犹如单簧管，很动听。

程序5-2

```
10  FOR I=1 TO 10
20  FOR J=1 TO 8
30  READ X,Y:SOUNDX,Y
40  NEXT
45  RESTORE
50  NEXT
60  DATA 12,4,14,4,16,4,17,4,19,4,21, 4, 23, 4, 24,4
```

需要注意的是：SOUND 语句用直接命令方式执行，或者用在IF THEN 语句中，虽能使主机发声，但不能同时输出音频信号，如在同一程序行中几个SOUND 语句连用，则必须用两个“：”号作分隔，例如：

```
10  SOUND20,4::SOUND30,4
```

如程序中有一系列前后相衔接的 SOUND语句，则每一语句的结束要加“：”号，例如：

```
10  SOUND20,4:
20  SOUND30,4:
```

否则，虽主机发声正常，但扩音器中却会丢失声响序列中偶数的声响。如果只要扩音器发声，不希望主机发声，可以键入：POKE-18672，4；如果要恢复同时发声，则可以键入：POKE-18672，37。

第四节　充分利用微型机的音乐功能

本节以APPLE-Ⅱ微型机为例，介绍微型机的音乐功能和编制音乐程序的方法，以便读者充分利用该机的音乐功能来演奏美妙的音乐。

一、微型机音乐功能原理

我们知道当一定强度的声频（16Hz～20kHz）电流流经扬声器时，就能产生一定响度和音调的声音。在一般情况下，声频电流越大，声音越响；电流频率越高、音调就越尖细；通电时间越长，发音持续时间也就越长。所以要使微型机奏乐，实质上是编制程序，以控制扬声器在一定的时间内通以一定强度的声频电流和电流频率，来产生不同的音调与节拍。

微型机里面并没有与各种音调相对应的声频电流源，只有时钟脉冲，它是高达几个MHz的高频脉冲，不能直接使扬声器发声，所以微型机奏乐实质上是利用时钟脉冲来得到能使扬声器发出各种音调的频率。解决这一问题并不困难，只要程序控制在每隔一定时间T给扬声器通一次一个（或几个）时钟脉冲，这就相当于给扬声器通以频率为F＝1/T的交变电流。只要频率落在声频范围内，扬声器即可发出相应音调的声音，而且音色也不错。控制通电间隔时间T的变化，就能产生各种音频的声音。控制扬声器连续通电的次数S，也就控制了扬声器发音的持续时间t（t＝S×T），这就是音乐的节拍。

现在进一步探讨一下，T和S如何按乐谱的音律和节拍来定值，根据有关音乐知识我们知道，对于十二平均律来讲，各音律名称与频率的关系如下所示：

音　律	C	D	E	F	G	A	B	C
频率(Hz)	264	296	332	352	396	445	498	528

音　　律	#C(♭D)	#D(♭E)	#F(♭G)	#G	#A
频率(Hz)	280	314	373	419	470

其中T(i)=1/F(i),通电间隔时间T可按上表定值。但是计算机还不能直接以这个T(i)值去控制音调。因为计算机是以一条条指令的执行来消耗时间的，所以可编一段汇编子程序来消耗这个T(i)时间，然后执行一条扬声器接通时钟脉冲的指令,循环执行这个子程序,扬声器就能发出相应音调来。

二、音乐程序简介

由于BASIC程序运行的速度太慢，只能使微型机发出频率很低的声音，所以不能用来编制发声程序，在音乐程序中只能用来编制有关的操作程序，发声子程序需用汇编语言编写，程序5-3即是它的清单。

程序5-3

```
9000—    A9 4C       LDA    #$4C
9002—    85 0E       STA    $0E
9004—    A9 90       LDA    #$90
9006—    85 0F       STA    $0F
9008—    A0 00       LDY    #$00
900A—    20 44 90    JSR    $9044
900D—    D0 16       BNE    $9025
900F—    20 44 90    JSR    $9044
9012—    F0 38       BEQ    $904C
9014—    AA          TAX
9015—    A9 10       LDA    #$10
```

```
9017—   85 13      STA   $13
9019—   88         DEY
901A—   D0 FD      BNE   $9019
901C—   C6 13      DEC   $13
901E—   D0 F9      BNE   $9019
9020—   CA         DEX
9021—   D0 F2      BNE   $9015
9023—   F0 E5      BEQ   $900A
9025—   85 10      STA   $10
9027—   20 44 90   JSR   $9044
902A—   A8         TAY
902B—   A9 0F      LDA   #$0F
902D—   85 13      STA   $13
902F—   A6 10      LDX   $10
9031—   48         PHA
9032—   68         PLA
9033—   48         PHA
9034—   68         PLA
9035—   CA         DEX
9036—   D0 F9      BNE   $9031
9038—   8E 30 C0   STX   $C030
903B—   C6 13      DEC   $13
903D—   D0 F0      BNE   $902F
903F—   88         DEY
9040—   D0 E9      BNE   $902B
9042—   F0 C6      BEQ   $900A
9044—   E6 0E      INC   $0E
9046—   D0 02      BNE   $904A
9048—   E6 0F      INC   $0F
904A—   B1 0E      LDA   ($0E),Y
```

```
904C—        60          RTS
904D—        24 18       BIT      $18
```

该程序中的902F—9038为子程序，末尾9038执行一条指令STX $C030，这是指机内扬声器通电一次。这段子程序的一个内循环执行时间为20μs左右，由X寄存器的值控制内循环次数N，内循环完毕才使扬声器通电一次，所以扬声器通电的时间间隔为N×20μs。这样我们只要按音律要求计算出T(i)值，然后除以20，取商的整数值存入寄存器X就可以了。

音乐的节拍是指一个音的发音时间长短，如进行曲一般演奏速度为每分钟120拍，从发音时间t＝S×T(S为通电次数）来看，由于t随节拍要求定值，所以S必须随音律而变。我们实际是让计算机控制扬声器通电次数，所以需求出S＝S(i)＝t/T(i)，把这个S值取整送入机内就可避免S>255，以至一个内存放不下，故将S除以8后存入内存。

上机操作的过程是这样的；先将程序5-3送入APPLE-Ⅱ微型机内存，再将乐谱送入首地址为904D的内存中去，然后键入CALL36864，就可听到计算机奏出的乐曲。

乐谱的送入可以通过编制一个BASIC程序来实施。程序清单见程序5-4。读者在这个程序末尾的DATA语句行中换上乐谱的音律与节拍即可。每个音律用1～7代表音律名C、D、E、F、G、A、B、并冠以前缀H，L、M。用M表示中音，H表示高音，L表示低音，如H_1表示高音C。在每个音律后，紧跟一个节拍数据，1为一拍，0.5为半拍，0.25为四分之一拍，2为二拍，依次类推。按上述方法依次将乐谱的音律和节拍值写入DATA语句行内，最后用两个0表示乐谱结束。注意休止符号也用0代表，但它后面必须紧跟一

个非0数据表示休止的节拍数，程序5-4要求最后一个数据必须是E，切勿忘记。在乐谱数据写完后，运行这个程序，机器将乐谱读入内存904D为首的内存区后即奏乐。您听完后，还可以改变演奏速度和音调，以得到满意的音乐效果。最后还可将“乐谱”存入磁盘保存（需用 BSAVE 命令）。

三、具有点播、储存和重放功能的APPLE-Ⅱ电子琴

前面已经介绍过，用机器语言可以使APPLE-Ⅱ微型机发出声音，如果把声音的频率和长短的控制交给键盘，那就使微型机变成一台简易电子琴了。程序5-5是一个电子琴程序，它可以使你在APPLE-Ⅱ微型机上弹奏音乐，也可以将音乐存入磁盘，还可以点播以前存入的音乐，并且有重放功能。

程序5-4

```
10   HIMEM: 36864
20   A=2^(1/6):D=SQR(A):T=1:P=1:M=2000
30   FOR  I= 36941 TO 37941
40   READ A$: IF A$="E" THEN  1000
50   IF A$="0" THEN N=100: POKE I,0: GOTO 130
60   IF LEFT$(A$,1)="H" THEN P=1
70   IF LEFT$(A$,1)="M" THEN P=2
80   IF LEFT$(A$,1)="L" THEN P=4
90   C=VAL (RIGHT$ (A$,1))
100  N=29*P*(A^(8-C))
110  IF C<=3 THEN N=N/D
120  POKE I, INT(T*N)
130  I=I+1: READ B
140  POKE I, INT (M*B/N)
```

```
150   N=0: NEXT I
1000  CALL 36864
1100  INPUT "LISTING AGAIN?(Y OR N)";Y$
1200  IF Y$="N"  THEN END
1300  INPUT "MODIFY VELOCITY? (T OR N)";Y$
1400  IF Y$="N" THEN 1700
1500  INPUT"VELOCITY?(ENTER>8 TO GO SLOW, <8
      TO RUN FAST)";V
1600  POKE 36908,V
1700  INPUT "RISING TONE?(Y OR N);Y$
1800  IF Y$="N"  THEN 1000
1900  INPUT "RISING NUMBER?(1TO7)";G
2000  RESTORE
2100  T=1/A^G:GOTO 30
3000  DATA M5,1, M3,.5, M5,.5, H1, 2, M6,1, H1,.5,
      M6,.5, M5, 2,M5, 1,M1, .5,M2,.5,M3,1,M2, .5,
      M1,.5,M2,3.5,0,.5
4000  DATA M5,1,M3, 5,M5,.5,H1,1.5,M7,.5,M6,1,H1,
      1,M5,2,M5,1,M2,.5,M3,.5,M4,1.5,L7,.5,M1,3.5,
      0,.5
5000  DATA M6,1,H1,1,H1,2,M7,1,M6,.5,M7,.5,H1, 2,
      M6,.5,M7,.5,H1,.5,M6,.5,M6,.5,M5, 5, M3,. 5,
      M1,.5,M2,3.5,0,.5
6000  DATA 0,0,E
```

运行程序5-5前，请先接好驱动器，然后输入这段程序。本程序的音乐弹奏键分别由以下几键代替：

ESC	Q	W	E	R	T	Y	U	I	O	P	RETURN
5̣	6̣	7̣	1	2	3	4	5	6	7	1̇	2̇

弹奏完一段乐曲之后，如果您想把这段乐曲存入磁盘，

准备以后点播，就可以按“S”键，计算机会在显示器上显示“NAME=？”，请你输入这段乐曲的名字，此后，计算机就会自动以此名将乐曲存入磁盘。

如果想点播音乐就按“A”键，计算机即询问你所要点播乐曲的名字，输入乐曲名后，这段乐曲就会放出，并可按“B”键来重放。这样您就有一台简易电子琴了。

程序5-5

```
0   ONERR GOTO 240
5   PRINT"     ,S,     SAVE"
6   PRINT"     ,A,     READ"
7   PRINT"     ,J,     END"
10  FOR ML=864 TO 891: READ MC: POKE ML, MC:
      NEXT
20  DATA 0,115,172,97,3,174,97,3,232,208,253, 169, 4,
    32,168,252,173,48,192,136,208,239,206, 96, 3, 208,
    238,96
30  DIM R(77),D(500)
40  FOR I=1 TO 12: READ M,N:R(M-13)=N:NEXT
50  DATA 27,24,81,52,87,74,69,85,82,105,84, 125, 89,
    134,85,151,73,166,79,177,80,182,13,192
53  K=0
55  GET A$:IF A$="A" OR A$="S" THEN  73
57  IF A$="J" THEN END
60  A=ASC(A$)-13:K=K+1:D(K)=R(A): POKE 864,
      2:POKE 865,R(A):CALL 866
70  GOTO 55
73  INPUT "NAME=";M$
75  P1$="OPEN"+M$:P2$="APPEND" +M$: P3$=
```

```
    "WRITE"+M$:D$=CHR$(4)
90  IF A$="A" THEN 200
100 D$=CHR$(4):PRINT
110 PRINT D$;P1$:PRINT D$;P2$:PRINT D$;P3$
120 FOR J=1 TO K
130 PRINT D(J)
140 NEXT
150 PRINT D$;"CLOSE"
160 GOTO 55
200 P3$="READ"+M$
205 K=0
210 PRINT D$;P1$:PRINT D$;P3$
215 K=K+1
220 INPUT D(K)
230 GOTO 215
240 FOR L=1 TO K-1:POKE 864,2:POKE 865,D(L):
    CALL 866:NEXT
250 GET A$:IF A$="B" THEN 240
260 GOTO 55
```

第五节　使微型机具有重播功能

程序5-6是用6502机器语言编写的程序，能将录音机的声音存储在APPLE机的内存中，需要时再将它调出来，使APPLE机重播输入的声音。

程序5-6

```
*300.37F
0300—  A2  00  A9  20  8D  14  03  AD
0308—  60  C0  29  80  48  A0  06  88
```

```
0310—  D0  FD  EE  00  20  D0  03  20
0318—  2A  03  68  C5  01  F0  E8  85
0320—  01  8D  30  C0  20  2A  03  4C
0328—  07  03  EE  13  03  D0  03  EE
0330—  14  03  AD  13  03  8D  3F  03
0338—  AD  14  03  8D  40  03  8E  5F
0340—  20  C9  90  F0  01  60  00
0350—  A9  20  8D  59  03  A0  0A  AE
0358—  00  90  86  01  88  D0  FD  A0
0360—  0A  CA  D0  F8  A6  01  F0  03
0368—  8D  30  C0  A0  07  88  D0  FD
0370—  EE  58  03  D0  E0  EE  59  03
0378—  AD  59  03  C9  90  D0  D6  60
```

地址0300～0347这一段是存储声音的程序。声音存放的地址为＄2000～＄8FFF。在这个程序中，＄30E单元中的数越小取声音的速度越快（如果这个数为＄00，那么它等效于＄FF），声音失真越小，但内存中能存放的声音也就越少；反之，这个数越大，取声音的速度也越慢，声音失真越大，内存中能存放的声音就越多。因此，这个单元中的数不宜太大也不宜太小，一般取＄5～＄8。

将声音放出来是利用地址350～037F这一段程序。＄356，＄360，＄36C中的数是控制放音速度的。调整这几个单元中的数值，可使放出来的声音不致于失真。

具体操作步骤如下：

先把录音机与主机连结好，将程序5-6输入主机，并且将一盘歌曲磁带放入录音机，按下“PLAY”键，然后在主机的监控状态下键入300G并回车。等到主机发出“嘀”的一声，同时显示器上重新出现“*”号，表明声音存储完

毕，便可关掉录音机。当在主机上键入350G并回车后，即可听到从主机的扬声器中放出刚才输入的音乐。

你也可以说几句话，用录音机录在磁带上，然后利用程序5-6存入主机。这样，便可随时听到微型机“复述”你的话了。

第六节　将电视机改装为显示器

我们可以将电视机改装为微型机的显示器。电视机与专用显示器相比较，不仅市场价格较低，而且功能较强。随着彩电的逐渐普及，黑白电视机闲置了不少，不妨将它改装成显示器。下面以LASER-310型微型机为例介绍如何改装电视机。

虽然LASER-310机本身提供了可直接接至电视机的射频信号，但由于射频信号经过电视机高频头及通道放大，容易受电视信号或其它高频信号的干扰，使显示屏上产生网状斜条，清晰度受到一定影响。此外LASER机的BASIC语言中有一个发声语句(SOUND)，很多人围绕这一功能编制了一些应用程序，例如使LASER机变成一台电子琴等。然而遗憾的是，该机的“发声器官”只是一个压电陶瓷片，发出的音量弱而音调尖，听起来象蟋蟀叫。

用改装后的电视机作LASER机的显示器可以解决显示和发音这两方面的问题。

一、显示部分

从根本上解决显示的清晰度问题，必须使用LASER机的视频输出信号，并将它直接送到电视机的视放级，但电视

原来没有视频信号输入插口，需对有关部分稍作变动，如图5-9所示。

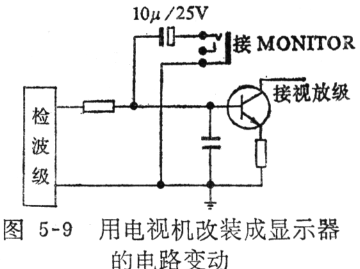


图 5-9 用电视机改装成显示器的电路变动

从电视机的视放级的输入端引出两线，信号端通过隔直流电容（10μF 25V）接到插座，另一端为接地端。在LASER 机的 MONITOR（监视器）输出电缆上接上插头，注意地线要对应。当插头插入插座时，即把计算机的信号加到了预视放级，此时，因不使用高频部分，不需要调节高频旋钮。经过这样改装后，屏幕字符和图形 都 十 分 清晰，这种措施能减轻上机人员的眼疲劳感。

二、发音部分

电视机的音频放大器能输出1W左右的功率，将LASER机的音频输出信号加至电视机低放电路的输入端，可从电视机的扬声器中得到较大的声功率，大大改善了 LASER 机的声音效果。具体方法如下：

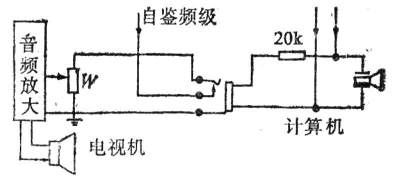


图 5-10 用电视机放大主机音频信号的电路变动

从 LASER 机的压电陶瓷扬声器两接线端引出两根线，其中一根串接上 20k电阻，然后将两线接入插头， 并在电视机上装上插座， 如 图5-10所示。

当插头插入插座时，即切断来自鉴频电路的信号，并向音频放大器输入计算机的音频信号，调节电视机的音量电位

器W即可实现对计算机声音信号的大小控制及取舍，当拔去插头后，电视机的伴音功能复原。

LASER-310机的扬声器在机内主机板下面，旋开主电路板的固定螺丝，掀开该板即可看到。

第七节　将录音机改装为外存储器

用普通家用录音机代替微型机的专用磁带机作外部存储器，是一种十分经济实惠的方法，已被许多学校所采用。但是实际使用的过程中也出现了一些问题。例如，由于各种型号的录音机输出电平不同，在向LASER微型机的主机输入程序时，即使反复调节音量电位器，也常会出现“LOADING ERROR”，有的录音机则根本无法进行正确的存取，这样既耽误了时间，又给操作者带来了很大的不便。本节将以LASER机为例，介绍用监视录音机输出电平来控制程序准确存、取的方法。实践证明，这种方法具有制作容易、成本低廉、效果明显等优点。有了这一装置，你就可以不必担心使用录音机作外存储器时程序存取的麻烦了。

一、磁带机存取程序的原理

将存储在磁带中的程序输入计算机时，数据从录音机耳机插孔送到LASER机的音频信号输入端，由计算机内的接口电路将音频信号(2400Hz表示逻辑“1”，1200Hz表示逻辑“0”)解调成串行的数据流(0、1两种电平)，从而拼装、转换还原成并行数据，其音频信号的振幅需要接近$2V_{pp}$，这样计算机才能正确地加以处理，否则，音频信号过大或过小都会使计算机误处理而不能识别，根据这一原理，我们不

仿制作一个简单的电平监视器来监视放音信号。

二、磁带机输出电平监视及其改装

这种电平监视器实际上是个峰值电压表，它的电路如图5-11所示。其中二极管D_1和D_2可用普通的2CP硅管，正向电阻越小越好。C_1和C_2可用小型瓷介电容。这里我们用万用表的10V直流档来作电压表。

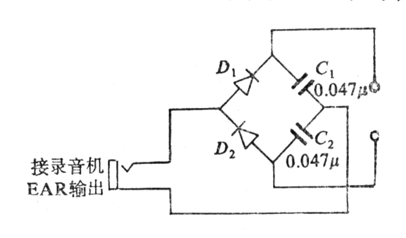


图 5-11 磁带机输出电平监视电路

由于二极管管压降和万用表输入内阻的不同，计算机允许接受的电平也有一个范围。使用时，可调整录音机音量，使万用表上的读数指示在$2V_{pp}$至$3V_{pp}$之间。若调入转储成功后，记下正确的电压读数，以后使用时便可照此操作。若有条件，也可以用一块录音机专用的袖珍电平表作为表头，装置成一个专用监视器，放在主机旁边，使用起来当然更为方便，并能随时监视计算机寻找调入的全过程。

有的国产收录机，仅有线路输入（LINE IN）而没有话筒输入（MIC IN），以致无法通过录音方式得到合适的放音电平，这时就要对录音线路动一点“小手术”。

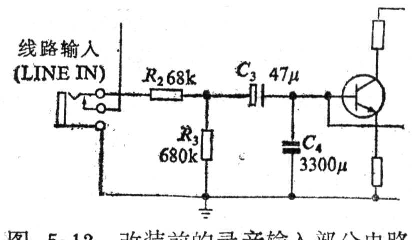


图 5-12 改装前的录音输入部分电路

例如：沪产单录机“美多 CP6010”，其录音输入部分的电路如图5-12所示。

按照图5-13的电路

进行改装后并不影响原机录放音乐的功能，只是使线路输入改成了话筒输入，且获得了合适的输出电平。

比较图5-12和图5-13中的两个电路后看到，改装电路中仅将R_3换成10k，并接在R_2之前。R_2的阻值应根据所改装的录音机进行调整（本例选择为2.7k）。调整工作可以这样进行：将一段程序存储进磁带，然后向计算机调入，放音时用上述监视器测量，若能通过调节音量电位器达到合适电压且调入成功，就可固定R_2的阻值，否则改换其他阻值再次试验。若你的收录机输入端虽有话筒输入（MIC IN）插口，但多次存取从未成功，也可用此方法试一试。若经过上述试验还不能成功地调入程序，可用下面方法：（1）清洗磁头或更换磨损的磁头；（2）选用新的优质磁带；（3）用干电池代替交流电源。

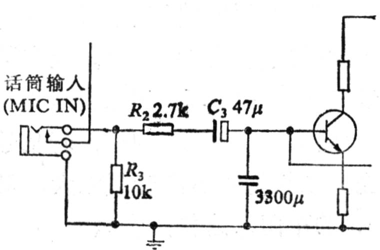
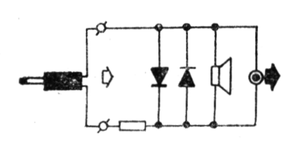


图 5-13 改装后的录音输入部分电路

在没有电平表头的情况下，也可以用发光二极管和扬声器来制作电平指示器。这里介绍一个成本很低的指示电路，可以从发光二极管的发光程度上，或从扬声器输出的音量上判断其输出电平是否符合微型机要求。图5-14是该简易装置的电路图。


图 5-14 电平指示器

制作这个指示器所需的全部器件包括一个3.5mm插头、两个发光二极管、一个电阻和一个小喇叭。两个发光二极管反向并联。喇叭用作监视器，如果喇叭发出两个非常清晰的

单声，则表示录音机是在发信号；如果喇叭发出轻轻的嘶嘶声，则表示录音带的内容是在两个程序之间。

指示器的插头插入录音机的耳机插口，磁带接口的输入经同轴插座也可连在上面。大多数微型机的录音带接口输入信号电平应达$2V_{pp}$。当录音机提供的信号约为这一电平时，指示器内发光二极管闪烁，若信号电平太高，发光管会连续发亮。如果嫌喇叭的音量太大，可在扬声器上串一个阻值为100Ω的电阻，这样，音量大小就适中了。

第八节 磁带机使用技巧

磁盘驱动器固然使用方便，但价格太贵。磁带机的速度、灵活性和可靠性虽然都不如磁盘驱动器，但它的价格便宜，且可以用录音机代替。因此，在一定的范围内，有着它的适应性。它可以用来储存程序及资料，并可以和随机存取存储器做各种信息交换。大多数学习机都有磁带机输入、输出插孔，并有一套关于磁带机的命令，因此，掌握磁带机的使用技巧是很有必要的。

一、使用磁带机的准备工作

1. 对磁带机和磁带的要求：一般的录音机都可以作为磁带机，有的需稍加改装（详见第七节）。作为磁带机的录音机应：

（1）磁头干净，马达转动正常；

（2）有话筒插孔和耳机插孔；

（3）有稳定的电源；

（4）有计数器。

如有专用磁带机当然更好。

磁带应尽量选用高质量的磁带，因为磁带上的任何一点缺陷，都可能使程序录不上。

2．磁带机与主机的连接：磁带机与主机之间是通过转录线连接的。在主机的后部，有一个输出插孔，标有“OUT”字样，将它与磁带机的“MIC”（即话筒）插孔相连；主机后部还有一个输入插孔，标有“IN”字样，将它与磁带机的“EAR”（即耳机）插孔相连。如图5-15所示。

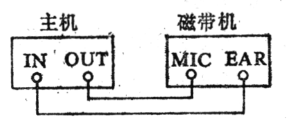


图 5-15 磁带机与主机的连接

二、与计算机交换资料

1．假定主机内有一个BASIC程序需要储存，可按如下步骤操作：

（1）将磁带倒至尽头，调好音量；

（2）在主机键盘上键入“SAVE”命令；

（3）同时按下录音机的录音和放音键；

（4）让录音机走带几秒后，按下微型机的回车键。

2．假定磁带内存有BASIC程序，你要把它调出来，可按如下步骤操作：

（1）将磁带倒至程序开头处，调好音量（第一次调出时可调至偏低音量的三分之一处）；

（2）在键盘上键入“LOAD”命令，暂不回车；按下磁带机的放音键；

（3）按下微型机的回车键。

在1和2的操作中，如果顺利的话，按下回车键后将会

出现这样的情况：光标消失；先听到一声“嘟”，说明已开始操作；过了一段时间后（这段时间视程序长短而定），又听见一声“嘟”，说明操作完毕，可按下磁带机停止（STOP）键。

在实际操作中，往往没有这么顺利，可能会有以下几种情况出现：

（1）产生信息“SYNTAX ERROR”；

（2）没有任何反应

（3）产生信息“ERR”或“ERRERR”。

如果遇到情况（1），只要按照以上的步骤重做一次。如果遇到情况（2）或（3），那么必须按CTRL-RESET键退出，调整音量，（第一次调试时，一般将音量由低向高调整）再重新操作一遍。出现这两种情况时，系统有时被锁住，将关机重来。

在输入输出过程中，音量是否适当是决定输入输出成功与否的关键。那么，究竟多大的音量适合于所用的机器呢？需要不断地试验，成功时记下音量旋钮所在处，以免下次再调。若使用上一节介绍的输出电平监视装置，需要记住存取的电平。

三、APPLE-Ⅱ微型机的有关指令

1．SAVE与LOAD：一般的BASIC程序都可以用SAVE命令存储到磁带上，用LOAD命令从磁带上调出，这两个命令后都不能带程序名，这是与其他一些计算机不同的。

2．STOP与RECALL：STORE命令可以把指定数组的所有元素存入磁带上。执行该命令时，录音机必须处于录音状态。RECALL 命令可把磁带上的数组元素读入指定数组

中。RECALL使用的数组名可以不同于STORE使用的数组名。但是要注意，存在磁带中的数组维数以及行列数必须和RECALL 时的数组维数以及行列数相同，否则将会产生 混乱，RECALL用的数组维数和最大下标值可以大于STORE所用的数组维数和最大下标值，多余的元素补 0 ，这时可能出现ERR，但不影响程序运行和数据。其格式为：

［行号］ STORE数组名

［行号］ RECALL数组名

在使用 STORE 和 RECALL 之前，应对各自的数组用DIM加以定义。

还要注意：字符串数组储存时，得先用 ASC 函数转 换成数值才可以储存。同样，在 RECALL 时也不能使用字符串，可以把读回来的数值用CHRS函数转换成字符串。

这两个命令可以在程序中使用，该命令的执行不影响程序本身。但是这两个命令并不能控制磁带机的开关，因而在程序中，在执行这两个命令的前后均应给出提示，由操作者控制磁带机。

3. W(WRITE) 与 R (READ)：这两个命令是用在监控系统内的，意思是写和读。请看下面这个命令：

（1） *D0000.FFFW

（2） *300.800R

第一个命令意思是：通知监控系统把由十六 进 制 位 D000开始到FFFF为止的内容存入磁带内。

使用该命令时一定要注意：

（1）必须确定磁带机一切正常，处于录音状态，磁带已正常转动，并转到有磁性的地方。

（2）操作时，按下回车键之后，机器要等十秒才开始

输送资料。只有当该工作完成后，才有“嘟”的一声，并给出提示符号“*”，光标又出现。

（3）如有错误，给出信息ERR。

监控系统输送资料速度大约为每秒210个字。

第二个命令表示把资料从磁带上读入内存，具体位置是十六进制的300到800。该命令执行时，磁带机应处于放音状态，且音量适当。

4．把造型表保存在磁带上和调入造型表：我们精心设计了一个造型表，可以把它保存在磁带上。在存入之前，必须先记下造型表在内存的首地址和尾地址及长度，均以十六进制表示。例如有个造型表首地址为6000，尾地址为600B，长度为B，如果将其储存，可以在监控中使用命令：

*0：0B 00回车

*0.1W 6000 600BW 回车

第一个命令是0～1单元中送入造型表长度；第二个命令是把数据写上磁带的两个命令。此时磁带机应处于录音状态。

SHLOAD 命令可以将存在磁带上的造型表调入内存中紧挨着HIMEM：的那个区域里，然后再把 HIMEM：指针自动移到造型表首地址那里，因而造型表被保护起来了。

当你的磁带上录有不少程序时，你就应该对它们进行编辑了。建议：

（1）把常用的程序录在前面，这样调出时方便；

（2）适当分类，将 BASIC 程序、机器语言程序、数组元素与造型表分开储存，以免混乱。

（3）在存储程序时，应记下记数器的数字，并编造目录，放在磁带盒里。这样便于在调出时查找。

（4）为了便于调出时查找，可以在每两个程序之间，存储一小段音乐程序。用录音机代替磁带机时，也可以在两程序之间录上一段话，用语言说明程序名。

（5）磁带应妥善保存，以免失去上面的程序。必要时，可以采取防抹措施。

第九节　用磁带存取LOGO程序

人们一般用磁盘来存取LOGO程序，然而不是所有的学校都能购置多台磁盘驱动器的，下面介绍用磁带存取LOGO程序的方法，从而使有磁带机的学校能开展LOGO语言教学。

一、用磁带存储LOGO程序的步骤

1. 键入命令ED ALL↙，将内存中的全部过程调入LOGO编辑区。

2. 键入 CTRL + G ，使机器退出编辑状态。

3. 查出程序的首地址和末地址。编辑区的首地址是$2000，这也是存储程序的首地址，程序的末地址随程序（过程和参数）的长短而变化。APPLE机内存中有两个单元是专门用来存储程序末地址的，其中156（十进制）单元存放程序末地址的低位字节，157（十进制）单元存放程序末地址的高位字节。用·EXAMINE 156↙和·EXAMINE 157↙命令可以查出末地址的低位和高位字节数。例如键入：

·EXAMINE　156↙

计算机若显示出：

RESULT：83

然后键入：

·EXAMINE 157

计算机若显示出：

RESULT：43

这说明程序的末地址是 43×256+83（高字节数×256+低字节数），要想求出末地址的十六进制数，可先把高字节数43变成十六进制的＄2B，再把低字节的83变成十六进制的＄53，最后合并成＄2B53，即为十六进制表示的程序末地址。

4．键入命令·BPT↙，让计算机进入监控状态。

5．用存带命令存入首地址（＄2000）到程序末地址间的全部内存内容。命令形式是：

首地址·末地址W（暂不按回车）

注意：首地址和末地址均用十六进制表示，且不加＄符号。例如，当末地址为＄2B53时，则键入的命令为：

2000·2B53W （暂不按回车）

6．将磁带机MIC插孔与计算机后边的OUT插孔连接起来。启动磁带机，记住计数器起始数值，置录音机为录音状态后，按下计算机的回车键，听到"嘟"声则存带结束。键入 CTRL + Y ↙，使计算机回到LOGO系统。

7．登记存入内容，磁带的位置（始、末数字）、在内存中的末地址的十六进制数值，以及末地址的高位字节和低位字节的十进制数值。这是今后读取程序时所必要的。

二、把磁带上的程序调入计算机的步骤

1．在LOGO系统下键入·BPT↙，命令使计算机进入监控状态。

2．用读带命令读入磁带相应位置的内容，键入命令形式如下：

首地址·末地址R（暂不按回车键）

例如：若末地址为＄2B53，则键入：

2000·2B53R

在按回车键之前，先要把磁带机的EAR插口与计算机后边的IN插口连接，再把磁带调到要读的LOGO文件的起始位置（看计数器值），按PLAY键置磁带机于放音状态，接着按计算机的回车键。当听到“嘟”声，表示读完。若显示屏上出现“ERR”，可能是由于音量不合适或存入程序时没有避过磁带前边的引带。

3．当程序调入后，键入 CTRL ＋ Y ↙，使计算机从监控状态回到LOGO状态。

4．把文件末地址的高位字节（十进制）和低位字节（十进制）分别送入计算机内存的157和156单元。例如末地址为＄2B53，则高位字（十进制）为43，低位字节为83，所用命令为：

·DEPOSIT 157 43 ↙

·DEPOSIT 156 83 ↙

5．用不带参数的ED命令对调入程序进行编辑，即键入：

ED ↙

这时屏幕上可能显示一些杂乱无章的亮方块。

6．按 CTRL ＋ C 键，退出编辑状态，这样调入的内容就编辑好了。

最后需要说明一点：这里介绍的LOGO系统磁带和程序存取方法都是在64k的APPLE-Ⅱ微型机上通过的。

第十节　磁盘使用技巧

一、在普通驱动器上使用双面磁盘

双面磁盘的正反两面都能使用，但是在普通驱动器上只能使用一面。这是因为驱动器左侧有一个光电写保护开关，磁盘保护袋写保护孔开放时才能往磁盘上写入信息。当磁盘反面插入驱动器时，磁盘保护袋左侧是封闭的，相当写保护孔封闭了，无法写进信息。另外，磁盘袋反面左侧也没有检索孔，所以无法使用磁盘反面。

但是，如果将$5\frac{1}{4}$英寸的磁盘加以改进，便可在普通驱动器上使用双面磁盘了。这样可大大提高磁盘的利用率。现将磁盘的改制方法介绍如下：

写保护孔

检索孔

图 5-16　磁盘示意图

在与磁盘袋原写保护缺口对称的地方再开一个写保护缺口，在磁盘袋反面与原检索孔对称的地方再开一个检索孔，位置如图5-16所示。

因为磁盘是很精密的外设部件，改装时一定要注意：开写保护孔要用锋利的小刀，盘下垫上平滑坚硬的平板，小心地用尺比着刻出一个缺口；开检索孔要把一个光滑的薄长条铜板小心从中心圆插入保护袋，要插到底，然后用打孔器轻轻地反复旋转，直到打出圆孔，这样就可以正反两面使用了。

二、一种充分利用磁盘空间的方法

这里介绍一种在APPLE 微型机上能够充分利用磁盘空

间的方法。

在用 INIT 指令将磁盘格式化时，APPLE-Ⅱ微型机会将DOS 自动存入磁盘内。其实，这是一种很大的浪费。因为只要拥有两三张能够存取DOS的系统盘就足够了，没有必要在每张盘上都用一定空间来存放DOS（当然，这样的确方便一些）。所以最好是在用 INIT 指令将磁盘格式化时，不要将DOS 存进去。按照下列方法可以不将 DOS 存进去：

```
]POKE42344,76
]POKE44723,4
]POKE46922,96
```

起动DOS,作上述修改后INIT空白磁盘，这时INIT的磁盘不但不含有DOS,连HELLO 程序也不存入。还原时可作如下修改：

```
]POKE42344,32
]POKE44723,12
]POKE46922,173
```

上述方法是针对空白磁盘的。对于已经存入资料的磁盘，运行程序5-7，同样可以达到除去DOS的作用。

程序5-7

```
140 PRINT: PRINT "INSERT DISK PRESS RETURN:":
    GET A$: IF A$<> CHR$(13) THEN 260
160 POKE 47084,17:POKE 47085,0
170 POKE 47083,0:POKE 47091,0
180 LOC=10000:POKE 47088,LOC—INT (LOC/256) *256:
    POKE 47089,INT(LOC/256)
190 POKE 768,32:POKE 769,227:POKE 770,3
200 POKE 771,76:POKE 772,217:POKE 773,3
210 POKE 47092,1:CALL 768
```

```
220 POKE LOC—60,255:POKE LOC+61,255:POKE LOC+
    64,255:POKE LOC+65,255
230 POKE 47092,2:CALL 768
240 VTAB 3: PRINT: CALL —958
250 PRINT: PRINT "DONE" : PRINT "UPDATE ANOTHER
    DISK?(Y N):": GET A$: IF A$="Y" THEN RUN
260 VTAB 20: END
```

用上述方法可以使一张磁盘多出33个磁区的使用空间。

三、废磁盘的利用

在使用$5\frac{1}{4}$英寸软磁盘的过程中，有时由于在目录磁道（$11道）上重写了其它数据，因此破坏了目录磁道原有的信息，这称为“软损伤”。有时由于磁盘被弯折或触摸了磁盘表面，或驱动器的磁头受伤，因此使目录磁道所在部位受到物理损伤，这称为“硬损伤”。上述故障使得磁盘既不能载送文件到机器内存中去，又不能把机器内存的程序文件存储到磁盘上，造成了死盘或废盘。如果目录磁道仅仅受到软损伤，只要启动DOS3.3盘，按INIT的步骤重新将磁盘格式化，即可使其仍然具有运载和存储文件的功能。但是，硬损伤就不易“医治”了。下面介绍如何加工目录磁道（$11道）部分受物理损伤的磁盘，使之能继续存储信息和文件。

根据对DOS3.3中INIT和CATALOG等命令及相关子程序的剖析和实验，可以发现当DOS3.3存入机器内存后，有下列内存单元是存放有关目录磁道的数据的：

$AC01、$AE9F、$AEDC、$B293、$B3BC、

$ B4BC、$ B5FA、$ AEC5、$ AEC9。

还可发现下列内存单元的内容与磁盘格式化分成的磁道总数有关：

$ AEB5、$ B3EF、$ BEFE

原格式化盘片的$ 0-2磁道为DOS磁道，$ 11 磁道称为目录磁道，$ 12—$ 22 磁道、$ 3—10 磁道为文件磁道，总共为十进制的35道（实践证实新磁盘的磁道总数可扩展到十进制的41道）。目录磁道损伤的废磁盘，当然不能取$ 11、$ 0、$ 1、$ 2、$ 3 等磁道为目录磁道。这里取大于$ 11的附近数作目录磁道为好，磁盘格式化分成的磁道总数以十进制30道左右为宜，因为这样具有可靠性。如自设$ 14磁道为目录磁道，又设格式化磁道总数为十进制29道，这样使废盘避开了被损伤部分的$ 11道，使废盘在$ 14道的第$ 0段存放磁盘内容编号表，在$ 1—$ 15 磁段存放文件名及有关信息，这样，废盘有可能起死回生。具体方法如下：

1. 启动DOS3.3盘，屏幕见光标后，键入NEW并回车，再键入CALL—151并回车，进入监控状态。

2. 修改机器内存有关单元的内容：

*AC01 ：14回车（目录磁道数以十六进制写）

*AE9F ：14回车

*AEDC：14回车

*B293 ：14回车

*B3BC ：14回车

*B4BC ：14回车

*B5FA ：14回车

*AEC5 ：50回车（4 *目录磁道数）

*AEC9 ：54回车（4 *目录磁道数 + 4）

＊AEB5：74回车（磁道总数＊4）

＊B3EF：1D回车（磁道总数即十进制29道）

＊BEFE：1D回车

3．按CTRL-C,回车。退出监控回到APPLE SOFT BASIC的状态后，从驱动器内取出DOS3.3盘，插入废盘片，关上驱动器门，再键入HELLO程序。例如

程序5-8

```
10  VTAB 10: HTAB 16: PRINT "HELL O!"
20  VTAB 16: HTAB 20: PRINT "STUD ENTS DISKETTES
    —1"
30  VTAB 20: HTAB 24: PRINT "1986.10.1"
```

4．键入INIT HELLO，回车。驱动器指示灯亮了，并发出正常旋转的声音。稍过一会儿，指示灯熄且屏幕又出现提示符和光标，说明格式化成功。如指示灯熄后没有出现提示符及光标，说明＄14道附近也有损伤，格式化失败，可另设其它磁道为目录磁道，将磁道总数减少后再进行试验。

值得注意的是，通常用DOS3.3格式化的磁盘，它们的目录磁道设在＄11道（即十进制的第17磁道），这种格式化的磁盘，我们称为“标准盘”。这里把某废磁盘的目录磁道设在＄14道（即十进制的第20磁道），它的格式化具有特殊的规格，因此以＄14道为目录磁道的磁盘与标准盘不能兼容。换句话说，两种盘片，必须在各自启动的情况下，才能在各自的范围内，执行DOS的内务命令，如CATALOG，LOAD、SAVE等。

四、用APPLE-Ⅱ微型机测定磁盘寿命

磁盘是有一定的寿命的，用到一定的时间就会丢失存在

磁盘上的文件和数据，为了保险起见，通常要另外拷贝一套备用盘。这样一来，虽然保险了，但却多占用一套磁盘，影响了磁盘的利用率。由此可见，如能测定磁盘的寿命，就可在它即将失效时再拷贝一套。现介绍一个程序，可用来判断磁盘的寿命。该程序是用6502机器码写的，其程序清单如下：

程序5-9

```
0300-A9  20  8D  C9  BD  A9  52  8D
030B-CA  BD  A9  03  8D  CB  BD  A9
0310-7A  8D  6E  BE  A9  03  8D  6F
0318-BE  A9  01  85  22  A2  00  BD
0320-2B  03  9D  00  04  E8  E0  28
0328-D0  F5  60  C4  C1  D4  C1  A0
0330-D2  C5  D4  D2  C9  C5  D3  BD
0338-A0  A0  A0  A0  A0  A0  A0  A0
0340-A0  D0  D2  C5  D3  C5  CE  D4
0348-A0  D4  D2  C1  C3  CB  A0  A0
0350-A0  A0  A9  30  38  ED  78  05
0358-48  4A  4A  4A  4A  09  B0  C9
0360-BA  90  02  69  06  8D  0D  04
0368-68  29  0F  09  B0  C9  BA  90
0370-02  69  06  8D  0E  04  CE  78
0378-05  60  48  4A  48  4A  4A  4A
0380-4A  09  B0  C9  BA  90  02  69
0388-06  8D  24  04  68  29  0F  09
0390-B0  C9  BA  90  02  69  06  8D
0398-25  04  68  4C  BE  8E
```

输入程序时，先用 CALL-151 命令使机器进入 监 控 状态，屏幕上出现监控提示符“＊”，从＄300单元开始输入。输入完毕后，退回DOS状态，在提示符“]”下，使用BSAVE

命令，将该程序存入磁盘，具体方法是：

键入：]BSAVE文件名，A＄300（起始地址），L＄90（地址长度）。

以后要运行时，可用BLOAD命令将该文件调入机器内存（如键入]BLOAD文件名），用]CALL768运行。然后将要测试的磁盘放入指定的驱动器中，这时如执行LOAD、CATALOG等与磁盘操作有关的命令时，在屏幕的第一行就会显示以下信息：

DATA RETRIES＝×× PRESENT TRACK ××

在等号的右边，出现一个十六进制的数。当该值小于＄10（十六进制数，下同）时，表示该磁盘还是比较可靠的；当该值在＄10～＄20之间时，表示应该建立备用盘了；当该值在＄20～＄30之间时，表示如再不建立备用磁盘就有危险了；当该值达到＄30时，驱动器读盘时可能会有杂音，这时该盘就要失效了。

第十一节　单面驱动器的改进

在APPLE-Ⅱ微型机系统中配置的驱动器都是单面驱动器。由于APPLE-Ⅱ的驱动器在读写磁盘时，不需索引孔进行定位，因此前面介绍了将双面软盘的写保护口对面开一个口翻过来使用，以提高磁盘的利用率。这里将介绍另一种方法，即把驱动器写保护电路加以修改，这样就不必将每张盘都开口了。由于驱动器的型号种类繁多，不能一一介绍，这里以全高度驱动器为例，至于其它型号的驱动器，其原理是一样的。

全高度驱动器写保护传感原件是一个微动开关。在不写

保护时，处于常开状态，贴上保护封签后，微动开关压下使信号断开。我们可以将此开关卸下，断开信号线，在驱动器外壳上安装一个双向开关，将信号线接在这个开关上。写盘时，使开关处于ON状态。保护时开关处于OFF状态。这样在机壳上控制读写保护状态，就再也不需要在盘上开口了。但要注意的是，开关处于ON状态时，既便贴了保护签，也会有写盘动作。因此，在不允许写盘时，必须将开关处在OFF状态。

半高度薄形驱动器的传感原件是一个光电管，原理同微动开关相同，只是改起来稍微比全高度的要麻烦一些。

第十二节　用磁带机运行磁盘程序

先分别介绍A、B、I、T类文件的意义：

A（APPLESOFT）类文件是浮点BASIC文件；

B（BINARY）类文件是二进制（即机器语言）文件；

I（INTEGER）类文件是整数BASIC文件；

T（TEXT）类文件是文本（即数据）文件，它是用OPEN命令打开，用WRITE命令写入的文件。

一、将A类文件从磁盘中转录到磁带中去的方法

首先把录音机（或磁带机）与主机连接好，并调整好音量开关，放入一盒空白磁带，把存有程序的磁盘放入第一号磁盘驱动器内，关好驱动器的门。然后启动计算机，当屏幕上出现闪动的光标后，键入CATALOG命令，即可看到文件目录，找到所需要转录的文件名，再按以下步骤操作：

1．］LOAD〔文件名〕↙（把程序从磁盘中调入内存）

2．］LI ↙（可看到程序已调入内存）

3．］SAVE（把程序从内存存入磁带）

先不要按回车键，记下录音机上计数器的号码，同时按下录音机的 RECORD 键和 PLAY键，然后再按主机的回车键。此时，屏幕上的光标不见了，大约过几秒钟后，听见“嘟”的一声，表示现已开始往磁带中存入程序了。程序存入后，又会听到“嘟”的一声，屏幕上出现闪动的光标。这时，按下录音机的STOP 键，记下录音机上计数器的号码。

二、运行磁带程序

1．把录音带倒回到原来的起始位置。

2．］LOAD先不要按回车键，按下录音机的PLAY键后再按回车键，屏幕上的光标将消失。过几秒钟后，会听到“嘟”的一声，且屏幕上出现了闪动的光标，表示程序已正确地从磁带中调入内存了。

3．按下录音机的STOP键。在键入 RUN 命令后，就可执行这个程序了。若在操作过程中出现 ERR 信息，则要关掉计算机，重新操作一遍。

三、将B 类文件从磁盘中转录到磁带中去的方法

先作好有关准备工作，启动计算机。若在键入CATALOG命令后，发现要转录到磁带中的文件属于B 类文件，则必须利用内存中的监控程序找出程序在内存中的起始和结束地址，并在监控程序下，把它存入到磁带中去。具体步骤如下：

1．］BLOAD〔文件名〕↙（把程序从磁盘中调入内存）

2．］CALL—151↙（进入监控程序）

3．* AA72.AA73↙（找出刚才用BLOAD调入的程序在内存中的起始位置）

（假如有）AA72—FD 07（则程序的起始地址为 07FD <16进制>，其中FD为低位地址，07为高位地址）。

4．* AA60.AA61↙（找出刚才用BLOAD调入的程序的长度）

（假如有）AA60—03 8E（则表示程序的长度为 8E03）

由以上假设，程序在内存中的结束地址为 07FD＋8E03＝9600<16进制>

5．记下录音机上计数器的号码，然后在监控程序下把程序存入磁带中。键入07FD.9600W先不要按回车键，同时按下录音机的RECORD键和PLAY键，再按主机的回车键，此时屏幕上闪动的光标消失。当闪动的光标在屏幕上再度出现时，就表示程序已正确存入到磁带中去了。现在可按下录音机的STOP键，记下计数器上的号码。

四、如何使用录有B类文件的磁带

要把刚才存入磁带中的程序调入计算机中运行，必须在监控程序下，用到刚才存入程序的起始和结束地址。先把磁带倒回到存入程序的开始位置，然后按以下步骤操作：

1．] CALL—151↙（进入监控程序）

2．* 07FD·9600R（把程序从磁带中读出）

注意，先不要按回车键，待按下录音机的PLAY键后再按回车键，此时屏幕上闪动的光标会消失。当程序调入结束后，屏幕上又会出现闪动的光标。此时按下录音机的STOP键，要运行这个程序，只要在监控程序下键入以下命令即可：

＊ 07FDG↙

对于有 48k 内存的APPLE机，用BLOAD命令调入的任何一个程序，都可用AA72·AA73和AA60·AA61 找到这个程序在内存中的起始地址和它在内存中的长度。当然，不同的程序在这四个单元中会显示不同的起始地址和不同的长度，这里的07FD和8E03只不过是举例说明的一个假设而已。

对于Ⅰ类文件，因计算机内存中没有整数BASIC的解释程序，要把这类文件存到磁带中去，就必须把INTBASIC文件同时存入到磁带中去，并且在从磁带中调出Ⅰ类文件时，也要同时调出INTBASIC文件。对于T类文件，要把它存入磁带中非常麻烦，而且一般来说也没有什么必要，这里就不介绍了。

第十三节　把微型机变成打字机

你可能要说，在计算机的键盘上按一个键，屏幕上就会显示一个字符，这不和打字机一样吗？

你仔细试试并作对比就会知道，它们并不完全一样。

首先，微型机在每次回车前所能接受的字符个数有限，一般是 255 个字符，多了不行；另外，每次回车后，它都以为刚才键入的是一条指令，经常给你个错误信息（？SYNTAX ERROR）。要是碰巧，内容中有微型机认识的保留字符，它还会瞎起劲，为你执行一番呢？例如，你要给外婆写信，打下了“GRANDMA”（外婆）这个词，但一按回车键，屏幕上方的字符都没有了。原来，计算机收到GR这两个起始字符，以为你要画图，从而进入了绘图状态。

能不能使计算机改变这种状况，使它变成一台打字机呢？

这就需要软件来帮忙了。先将程序5-10输入计算机并按回车键，这时计算机就变成了“打字机”，一切操作与真正的打字机一样，按下回车键的意义仅仅是换行，不把内容当作程序读入。一页写满后，逐行向上移，如果要结束打字练习，只需按CTRL-C即可。

要是文章较短，利用屏幕硬拷贝可以把内容用打印机打印出来。

程序5-10

```
100   GOSUB 1000
110   F=FRE(0): GET T$: IF T$=C$ THEN HOME:
      END
120   IF T$ = CHR$(21)THEN T$=CHR$ (PEEK (B
      (C)+PEEK(36)))
130   PRINT T$;:C = PEEK (37):GOTO 110
1000  TEXT:HOME:DIM B(23):C$=CHR$(3):REM C$
      IS CTRLC
1010  FOR I=0 TO 23:READ B(I): NEXT:RETURN
1020  DATA 1024,1152,1280,1408,1536,1664,1792,1920
1030  DATA 1064,1192,1320,1448,1576,1704, 1832,1960,
1040  DATA 1104,1232,1360,1488,1616,1744,1872,2000
```

第十四节　打印机使用技巧

一、利用打印机打印打字蜡纸

用打印机打印份数较多的资料，往往速度很慢，打印汉字时更是如此。若打好后用复印机印，成本又很高。为了解决这个矛盾，可以用打印机打印打字蜡纸，然后再油印。在

打印中需要注意以下三点：

1. 不要将色带取下。因为打印头是比较精密的器件，针孔与打印针的尺寸配合比较严格，直接将针头打在蜡纸上容易使针孔发生堵塞，而隔着打印色带一般不会发生这种现象。

2. 由于蜡纸较短，在打印蜡纸底部时，纸尾可能已经通过了打印机探纸器的位置，从而亮灯报警。解决这个问题可以在蜡纸的衬纸下沿粘一纸片，或用一张纸条，一头塞在探纸器下，另一头用重物压住，使之不随纸辊移动。

3. 将打印机给纸方式的板手推向后方，即纸辊磨擦走纸方式进纸。

二、延长打印机色带的“寿命”

一根色带用不了多久，就使打出的字颜色变淡了。用户往往丢掉这根色带，再去买一根新的。其实打印色带此时不一定是没色了，而是色干了，有两种方法可以延长它的“寿命”：

1. 将色带的上、下边倒一下使用，因为大多数打印机的针头并不是打在色带的中心，而是稍微偏一些，倒过来使用如同使用新色带一样。

2. 对于色干的色带，可以采取“上色”的办法，即上点15～20#的机油。因为上油之后，可以将干化的颜色润开。具体的方法是：将色带盒打开，用草棍沾少许机油在色带上星星点点地涂上。千万不要上多，否则打印结果将一团模糊。然后转动色带，目的是使油扩散均匀。最好隔一天再用。一根色带一般情况下可以上 4 ～ 5 次油。

三、西文汉字、不同字体交替打印

APPLE-Ⅱ在一般情况下处理不好西文汉字交替打印或同一行不同字体的打印。例如，大多数APPLE-Ⅱ的汉卡没有合适的造表符号，在汉字状态下，无法用大小合适的英文字符来构制美观的表格及其框架。下面介绍一个子程序，利用它可使微型机由汉字状态转入西文状态，然后再返回汉字状态，子程序清单如下：

程序5-11

```
4000  REM -BGDY
4010  POKE 1403,0
4020  PRINT D$;"PR#1"
4030  PRINT CHR$ (27);"J";CHR$(N);
4040  PRINT D$;"PR#0"
4050  PRINT:PRINT:PRINT
4060  PRINT D$;"PR#3"
4070  POKE 1403,1
4080 RETURN
```

程序中的D$ 已由主程序赋值为CHR$（4），4030语句中的“J”如无法直接键入，可用CHR$（106）代替，N为回转参数，由用户自己确定。程序5-11适用于打印轴具有倒转功能的打印机（如FX-800型），至于无倒转功能的打印机（如FX-100型），则需要将程序5-11的4030语句改为

```
4030 PRINT CHR$ (27); "3"; CHR$(N);A$
```

其中N为改变点距参数，由用户确定。N越小，两行之间的

距离越小，N为1时两行几乎重合。A $ 为多个“-”或英文字母的组合。

第十五节　计算机在选择和维修电视机、录象机中的应用

一、在选择电视机和录象机时的应用

因计算机有视频信号输出接口，有的还有射频信号输出接口，所以编制一些简单的程序，计算机就可输出多种图形信号，作电视信号发生器用，以鉴定电视机和录象机的质量好坏。以下为用于APPLE-Ⅱ微型机的图形程序。

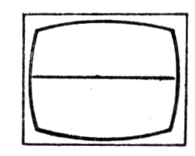


图 5-17　显示横线

1. 运行程序5-12，可使显示屏中间出现一条横线，如图5-17所示。

程序5-12

```
10  HGR
20  HCOLOR=3
30  HPLOT 0,96 TO 279,96
40  END
```

2. 运行程序5-13，可使显示屏中间出现一条竖线，如图5-18所示。

程序5-13

```
10  HGR
20  HPLOT 139,0 TO 139,191
30  END
```

图 5-18　显示竖线

3. 运行程序5-14，可使显示器中出现“+”字，如图

5-19所示。

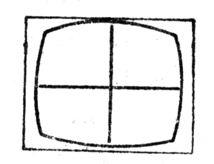
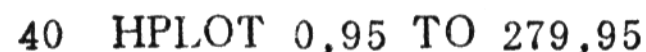
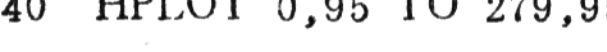
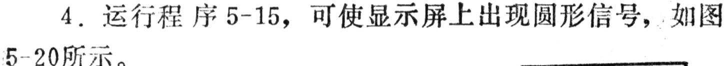


图 5-19 显示“十”字

程序5-14

```
10  HGR
20  HCOLOR=3
30  HPLOT 139,0 TO 139,191
40  HPLOT 0,95 TO 279,95
50  END
```



4. 运行程序5-15，可使显示屏上出现圆形信号，如图5-20所示。

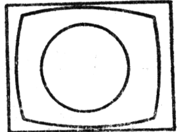


图 5-20 显示圆

程序5-15

```
10  CX=140:CY=96:FL=0
15  R=90
20  HGR2:HCOLOR=3
30  FOR T=0 TO 6.4 STEP .3
40  X=R* COS(T)
45  Y=R* SIN(T)
50  SX=X+CX:SY=CY-Y
60  IF FL=1 THEN 80
70  HPLOT SX, SY:FL=1
80  HPLOT TO SX,SY
90  NEXT T
100 END
```

5. 运行程序5-16，可使显示屏上出现方格信号，如图5-21所示。

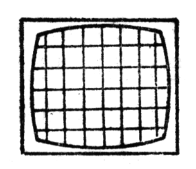


图 5-21 显示方格

程序5-16

```
10  HGR2
20  HCOLOR=3
30  FOR I=0 TO 6
```

```
40  J=34*I
50  HPLOT 34+J,0
55  HPLOT TO 34+J,191
60  NEXT
70  FOR K=0 TO 4
80  L=32*K
90  HPLOT 0,31+L
95  HPLOT TO 279,31+L
100 NEXT
110 END
```

二、在维修录象机中的简单应用

1．记录和重放故障的判断：用计算机给录象机输入一图形信号（用视频输入和射频输入均可），记录后再重放，观察图象情况并和记录时的图形进行比较，判断是记录还是重放故障。

2．时基误差的判断：用计算机给录象机输入一条横线信号，记录并重放，比较记录和重放时直线的变化情况，来判断时基误差过大产生的原因，是记录还是重放时产生的。

三、在维修电视机中的简单应用

1．电视机接收性能的检查：先用计算机给电视机输入一射频信号，根据图象的变化情况来判断接收性能，可大致判断是高频调谐故障，还是中频部分的故障。

2．同步故障的判断：用计算机给电视机输入一条竖线信号，观察竖线的变化情况，可判断出电视机图象是否稳定，行场是否同步。

用计算机输出各种图形信号，可看作“标准”信号，在

没有电视广播时使用，就更显得实用，尤其是在中、小学和一些小型电视维修部，可收到投资少，使用方便，充分发挥设备利用率的效果。

第十六节 软件的加密与解密

本节以APPLE机的软件为例，介绍加密与解密的方法。

一、普通软件的加密方法

1．对文件名进行加密保护：一般是在文件名中加入不显示字符，即隐蔽字。这些隐蔽字都是一些控制字，也就是：[CTRL] + [某字符]。这样，用CATALOG命令显示的文件名与软件中实际的文件名并不完全一样，从而既不能将文件调入内存也不能列出原程序清单、对程序进行各种操作，达到了加密的目的。

2．使列表命令失效：由于APPLE机上BASIC程序是在2049（$801）这个位置开始储存的，只要改动这个位置的值，便可使列表命令失效，具体步骤如下：

（1）键入型号0，然后按住CTRL键不放，再按任一字母键（除字母键P、X、C、S以外），最后按回车键。

（2）键入下列语句：

1 POKE 2049，1↙

或者键入：

CALL—151↙

＊801:1↙

CTRL—C↙

运行程序，然后列表，屏幕上将不断显示0，别人也就看不到你所设计的程序了。

3．利用死循环，阻止程序中断：在程序中加入如下语句：

程序5-17

```
0       ONERR GOTO 63999
1       POKE 1010,48:POKE 1011,3:CALL-1169
2       POKE 816,76:POKE 817,48:POKE 818,3
63999   GOTO 63999
```

这样，程序一旦启动，由于1语句将CTRT—RESET的控制转到$330处，2语句使$330处存有一条机器语言指令（JMP$330），从而陷入死循环。0语句与63999语句配合，使中断程序的企图（CTRL—C或CTRL—RESET）落空，从而程序得到了保护。

4．修改DOS系统。

5．采用非标准格式磁盘：

（1）整个磁盘采用特殊的格式，例如，可自成一运行系统，系统内不提供盘复制命令。

（2）在某一磁道采用特殊格式，例如把一些关键数据存储在特列格式磁道上。

（3）利用磁盘的备用磁道存储重要数据。

（4）在某一磁道的某扇区采用特殊格式，把程序的重要数据存储在这里。

6．采用密码口令：软件加密分硬加密和软加密两种。上面介绍的在软盘上采用非标准磁盘格式即是常用的一种硬加密。而软加密指加密手段不涉及任何物理变化的软件保护处

理。密码口令即是软加密的一种方法。

在软件编好之后，找一个不容易让人猜着的“口令”。这个口令的字符、内容可随意确定，字符数也不限，可以是某个物体的名称、某个正向或反向拼成的名词，或者是其他字符的集合。然后，在程序前面加上这样几个语句：

程序5-18

```
10  PRINT: HTAB 15: INVERSE :PRINT "PASSWORD?":
    NORMAL:INPUT PW$
20  IF PW$<>AP$ THEN 500: REM GOTO MENU
    OR END
30  REM CONTINUE WITH PROGRAM
```

在程序的20语句之前，你还要把已认可的“口令密码”赋值给AP$，可以用下面的格式：

15 AP$ = “×××××”

其中的“×××××”为你的口令。

程序5-18的语句号只是一个例子，你可以用适合自己程序的语句号。这样，你的程序就变为加上了口令保护的“保密”程序了。

在下面的程序5-19中，20语句POKE214，255保护了所有在这个盘片上的程序，甚至HELLO欢迎式开机程序也不受LIST命令作用。

程序5-19

```
10  D$=CHR$(4)
20  POKE 214,225
30  PRINT D$;"RUN BOOTSTRAP BASIC"
```

当你用这个盘片开机，即用PR#6 电源冷启动和热启动时，POKE命令就起作用了。事实上，只有另一个未加保护

的开机程序进入机器或关机时，它对所有顺序放入驱动器的磁盘程序才无影响。在保护状态下，除CATALOG、LOAD、RUN外，任何键盘 DOS 命令都将中止执行。

在RUN你的程序时，屏幕上会显示“PASSWORD”，让你输入口令字，并且显示出你键入的口令字。如果你不喜欢把口令显示在屏幕上，可以使用 GET 命令。GET 命令是APPLESOFT的一道叙述，它从键盘或其它设备上接受一个字符，但不会在屏幕上显示出来。使用GET命令比上述例子稍复杂一点。见程序5-20

程序5-20

```
10  PRINT: HTAB 15: INVERSE: PRINT "PASSWORD?":
    NORMAL
20  A$=" ":PW$=" "
30  FOR I=1 TO 5
40  GET A$
50  PW$=PW$+A$
60  NEXT I
70  IF PW$=AP$ THEN 110
80  HOME:VTAB 10:HTAB 10:INVERSE:PRINT "INVALID
    PASSWORD":NORMAL
90  FOR I=1 TO 500:NEXT I
100 GOTO 500:REM MENU OR END
110 REM CONTINUE WITH PROGRAM
```

其中，500语句没有列出，可以根据自己需要来确定，或转“菜单”，或转结束。30语句规定了口令字为五个，你可随意增减。80语句的作用是输入的口令字不对时，屏幕将告诉你：口令无效。执行到100语句，即转“菜单”或结束。第110语句为你的程序开始。

同前面一样，在程序的前面，要先把已认可的口令赋值给AP＄，不管这一句加在哪里，能先知道就行。

为了增加破译的难度，口令字串可使用一些非打印控制码，也就是这些键按下后，屏幕上不出现任何字符，但有时会发出声响，这些看不到的字符还可以用来做文件的名称。使用这些非打印控制码的口令或文件名，保密效果很好，即使有人看到程序清单，他也未必能马上破译。

当这些新加的语句与你的程序连接好以后，请拿出一片新磁盘片，插入磁盘驱动器，并且把HELLO欢迎式用 INIT 命令将磁盘格式化，然后用规定的文件名把上述程序存在盘上，这样就制成了一个不显示口令字的保密软件了。

二、简便解密方法

1．用解密程序使隐蔽字显露出来：程序 5-21 即是一种解密程序。将该程序用SAVE JEMI存盘保留。使用时，运行该程序后再键入CATALOG 命令，文件名中的隐蔽控制字即以反相形式显露出来。将待解密的程序调入内存后（不要运行），再键入 LIST，文件名中以及程序中的隐蔽字均以反相形式显露出来，不过，隐蔽字一般不能打印出来。

程序5-2.

```
10  FOR I=768 TO 795:READ V:POKE I,V:NEXT
20  POKE 54,0:POKE 55,3:CALL 1002
30  DATA 201,141,240,21,201, 136, 234, 234, 201, 128,
    144,13,201,160,176,9,72,132,53,56,233,64, 76, 249,
    253,76,240,253
```

2．用LOGO语 言系统盘找出文件名中的隐蔽字：先在DOS3.3系统下，用命令CATALOG列出目录，记下文件名。

然后引导LOGO语言系统盘，在提示符“？”下，再一次用CATALOG命令列出文件目录，后列出的文件名与前次列出的文件名相比，多出的字符就是文件名中插入的控制字，这些控制字为正相显示。

3．改换加密文件名：用DOS盘上FID程序在复制文件过程中换名，将含有隐蔽字的加密文件名改为自己重新命名的文件名。避开文件名中的隐蔽字重新复制一个文件后，即可进行程序的增、减、删、改等各种操作，也可打印出源程序清单。具体方法步骤为：

（1）用命令BRUN FID运行FID程序；

（2）选菜单项1，COPY FILES；

（3）指定源盘及目的盘的驱动器插口号和盘号；

（4）文件名用“=”泛指替代；

（5）屏幕上显示“DO YOU WANT PROMPTS？”（你要提示字吗？）时，键入Y↙回答。程序不用一个统称的卷名来代表每个指定文件，每读入一个文件后，显示文件名。这时你可用一个新文件名代替源盘上的文件名，换名后存入目的盘，不必顾及文件名中是否有隐蔽字。例如显示的文件名为“HELLO”，其中可能含有隐蔽字，假如文件真实名为“HELLO |CTRL—O|”（CTRL—O为隐蔽字），这时你可能键入新文件名“HELLO”（五个字符，不含隐蔽字），这样拷贝后的新文件名就是“HELLO”，不含有隐蔽字。当然你也可以给程序换其它的文件名。

4．无法直接找到文件名程序的解密：对无法直接找到文件名的解密途径，一是设法恢复CATALOG命令功能，二是可在程序调入内存的运行过程中，设法找到程序断点或人为使程序产生断点，从而可打印出源程序清单并将程序转入自

巳盘中。程序运行中，通常总要进行菜单选项、数据输入、查询、修改以及运行结果输出打印等处理，在这些地方可利用故意错输或误操作，使程序非正常中断，从而截获源程序。

对修改了第一条语句链指针、LIST只显示第一句的程序，可在监控状态下，查找被破坏了的链指针，予以恢复。具体步骤为：

（1）]CALL—151↙

*（转入监控状态）

（2）*800·87F↙

0800—00 00 08 00 0A……

（3）从804单元往后查找第一句句尾(为0 0)，其下一单元即为第二句的开始，为第三句的链指针。将链指针的单元地址（设为$081E）存入801和802单元（低位在前，高位在后）；

（4）*801:1E 08↙

（5）*CTRL-C↙

]（退出监控，返回APPLE SOFT BASIC状态）

此时即可键入LIST，从而获得源程序清单。

5．查找并恢复有关指针：许多加密程序除用破坏语句链指针使源程序清单不可LIST外，往往还同时用设置热启动及清除程序、设置死循环等方法使你难以用CTRL—C或CTRL—RESET中断程序，即使程序中断，也让程序在中断前清除掉。从而使你很难截获源程序、打印清单或转存获取该加密程序。典型的处理方法，除前面提到的外，还有：

```
0 POKE1012, 166:ONERR GOTO62534
:
```

⋮

⋮

62534 PRINT CHR$(4); "PR#6"

或62534 PRINT CHR$(4); "PRN××××"(××××)为文件名)

或62534 NEW

……………………

上述0语句中POKE 1012,166使按下CTRL—RESET时，计算机自动重新启动。0语句中ONERR GOTO 62534和62534语句配合，使中断时重新调入程序、引导磁盘或清除程序，使程序运行中难以出现断点，保护程序不易被截获。我们知道，NEW命令并未清除内存中程序，只是将与程序有关的几个指针恢复为初始值。而对于"PR#6"命令或运行新文件命令"RUN ××××"，只要驱动器中没有软盘，调不进新程序，内存中原有的程序是不会被冲掉的。此时只要再按一次CTRL—RESET键，即可切断驱动器，返回APPLE SOFT BASIC状态。如果程序已被"NEW"，查找并恢复有关指针后，即可获得源程序。

几个指针中，最主要的是程序区尾指针，它存放在$AF、$B0中。其它有关的几个指针有变量表首址（$69、$6A中）、数值表首地址（$6B、$6C中）、数组表尾地址（$6D、$6E中），其初值均同程序尾地址。查找和恢复以上几个指针，可先用前法恢复BASIC语句链指针后，用LIST命令查看源程序（此时切不可修改及增删语句），根据程序长度估计程序尾大致存放位置，记下最后一句用以核对所查得的程序尾。查找程序尾的方法是在监控状态下，找

到连续三个00单元，即为程序尾。将这三个00单元下一单元地址存入以上四个指针内（低位在前,高位在后），返回BASIC状态，即可获得源程序并可存盘。当然，查找程序尾需要掌握程序在内存中的存放方式，熟悉ASCII码，但在经过一般练习后是不难掌握的。

例如，在监控状态下找到程序尾为＄1E5C，那么，恢复各指针的操作如下：

1 E58—×× 00 00 00 ×× ×× ×× ××

* AF:5C 1E↙

* 69:5C 1E 5C 1E 5C 1E↙

* CTRL—C↙

] （返回BASIC状态）

6．对修改DOS系统的加密软件解密：有一些程序，即使在恢复了指针后，可以正常列出程序清单和运行，但往往也无法存盘。这是由于DOS被修改了的缘故。如直接调入未经修改的DOS，则会把内存中的程序冲掉，为此可将BASIC程序移到DOS装入过程中，不需使用的安全区域：＄4000～＄9600（只要程序长度不超过此范围），并要将零页指针保护起来，移入＄9560起的区域。调入DOS后，再将程序和零页指针移回即可。具体操作步骤如下：

（1）记下零页中程序尾指针（＄AF、＄B0）的值，并计算程序长度：

长度＝程序尾地址—＄800＋1

（2）移动程序，在监控状态下操作：

*4000<800程序尾地址M↙

当程序长度超过＄3800时，应分两次移动。先移动＄4000后的部分，再移＄800～＄3FFF的前半部分。

*7800<4000。程序尾地址M↙

*4000<800.3FFFM↙

（3）保存BASIC程序零页指针：

*9560<60.FFM↙（假定程序不超过＄955F）

（4）装入DOS

*C600 G↙

（5）取回零页指针

]CALL—151↙

*60<9560.95FFM↙

（6）移回程序

*800<4000.4000+长度M↙

（7）返回BASIC状态

*CTRL—C↙

]

然后即可把程序存盘，并可打印清单、运行该程序了。存盘时文件名可自行另取。

7．对隐取目录的软件解密：对有些不显示目录的软盘，运行程序5-22后再键入CATALOG，往往就可显示隐蔽目录了。

程序5-22

```
10  FOR I=833 TO 872:READ V:POKE I,V:NEXT:CALL
    833
20  POKE 24587,18:POKE 859,2:CALL 833
30  DATA 169,3,160,79,32,217,3,169,0,133,72,96,0, 0,
    1,96,1,0,17,15,101,3,0,96,0,0,1,0,0,96,1,0,0,0, 0,
    0,0,1,239,216
```

8．B类文件的解密：对加密了的B类文件，可用DOS盘

上的FID程序更名转录，予以解密。也可找出文件在内存中的存放起始地址和长度，重新命名存入磁盘中。B类文件存放起始地址可从＄AA72、＄AA73查出，长度可从＄AA60和＄AA61中查得。例如对内存中某一B类文件，进行如下操作：

```
] CALL—151↙
*AA72. AA73↙
AA72—06 08（设起始地址为＄0806）
*AA60. AA61↙
AA60—C8 01（设长度为＄01 C8）
*CTRL—C↙
] BSAVE [文件名]，A＄0806，L＄01 C8↙
]
```

即可将该B类文件转存于非加密盘。

9．有口令保护的软件解密：用一片未加保护的开机程序盘启动计算机，然后用LOAD命令把有口令保护的磁盘上的程序存入内存，再用LIST命令即可列出程序清单，找到口令字。

对于使用隐蔽字的口令，可用上面介绍的显示隐蔽字的方法。

第六章　自制实验用电源及辅助检测装置

第一节　接口电路实验用稳压电源

在前面介绍的较多实验中，自制接口电路的电源部分需要使用外接直流稳压电源。市售的稳压电源价格较贵，本节将介绍一种直流稳压电源的制作方法，这种电源的电路简单、自制方便、价格便宜而性能优良。

一、TDB0117集成稳压器简介

三端集成稳压器是七十年代发展起来的新型集成稳压器件。它具有稳压性能指标高、输出电流大、使用方便等特点，因而在各种电子仪器与设备中得到了极广泛的应用，是目前世界上线性集成电路中产量仅次于运算放大器的产品。

这里我们使用的TDB0117，是输出电压可以调节的三端集成稳压器，它的内部有近三十个晶体三极管和二极管以及阻容件所构成，来完成各种稳压和自动调节的功能。它比传统的稳压电路和输出电压固定式三端集成稳压器具有更突出的优点。TDB0117不仅稳压性能好，而且使用十分方便，只需外接一个电阻和一个电位器就可以获得连续变化的各种输出电压。器件内部还设有功能齐全的保护电路，在输入、输出及调整端

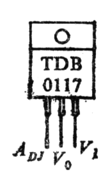


图 6-1　TDB0117的外形

连接正确的情况下，短时间的输出短路、过热等故障，一般都不会造成损坏。它的外形如图6-1所示。

它的外壳封装为 T0-220 型，图中V_I、V_0和 A_{DJ} 分别是稳压器的输入、输出和调整端。它的主要参数指标如下：

它们的主要参数指标如下：

最大输入输出电压差（V_I-V_0）$_{max}$40V

输出电压可调范围：1.25V～37V

最大输出电流I_{max} 1.5A

电压调整率S_v0.01%V

电流调整率S_I（$V_0\geqslant5V$）0.1%

纹波抑制比S_R（$C_{ADJ}=10\mu F$）80db

基准电压V_{REF} 1.25V

调整端电流I_{ADJ}50μA

温度系数T_C 1%

二、1.25V～24V/1A可调输出稳压电源

在TDB0117集成稳压器的V_0、A_{DJ}和地之间外接电阻R和电位器，就组成了基本的可调输出稳压电源。图 6-2 是它的原理图。

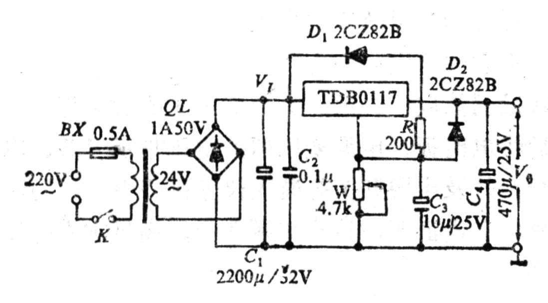


图 6-2 直流可调输出稳压电源

图中变压器次级输出24V交流电压，经桥堆QL全波整流和电容C_1滤波后加至集成稳压器的稳入端。除了取样电阻R和输出调节电位器W以外，C_2用于旁路高频脉冲干扰。在仪器设备中，当稳压器被安排在远离整流滤波电路时，就更有必要加入此电容。C_3用来改善稳压电源的纹波抑制特性，使输出电压的波动直接反馈至稳压器调整端。使用C_3之后，可将纹波系数提高15db。输出电容C_4兼有防止自激振荡的作用。

D_1、D_2是保护二极管。万一输入端对地短路，则输出电容C_4两端的电压将作为集成电路的反向电压加于V_0与V_I之间。有了D_1，就可以在输入端电压低于输出端时，使V_0-V_I箝位在0.7V。D_2可防止在输出或输入端短路时，C_3通过A_{DJ}端对集成块放电。

为了充分发挥集成稳压器的优良性能，保证稳压器的安全使用，TDB0117集成块上应装上散热面积不小于100 $(mm)^2$的散热片。散热片可用铜板或铝板自制。

在所有元器件焊接正确的情况下，稳压电源不需要作调整。接通电源并调节电位器W即可在V_0端得到1.25V～24V间连续可调的输出电压。如要试一下该电源的稳压特性，可在输出端接上一个可以调节阻值大小的大功率电阻器，使其输出电流为1A左右，用数字式电压表测出其电压的变化情况，然后计算出它的稳定度和内阻。纹波电压可用毫伏表直接测出，这些数据都应符合前面介绍的有关参数指标。

最后将电路板、电源变压器，电压指示表头、电流指示表头、指示灯及电源开关等装入仪器机壳中，就制成了一台稳压电源。实践证明，这台电源完全能满足本书介绍的接口实验需要。

使用时，特别需要注意的是，不能让集成块的输入输出

压差（V_I-V_0）超过其额定值，否则会造成集成块的永久性损坏。

第二节　三态TTL逻辑测试笔

在接口电路实验中，多次介绍到微型机输出电平和接口电路的电平变化。因此，电平的测量是十分需要的。本节将介绍一种制作容易、成本较低的三态TTL逻辑测试笔。如在微型机应用实验中使用它，将会给你的实验带来很大的方便。这种测试笔也可以用作检修数字电路及微型机电路的工具。

由于只使用了一块集成电路和少数分立元件，所以制作的测试笔体积小巧，只有钢笔那么大，使用起来十分方便，也便于携带。图6-3是测试笔的电路图。

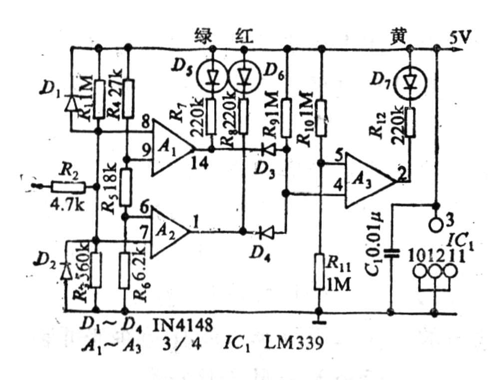


图 6-3　逻辑测试笔电路

电路选用了LM339组成电压比较电路，LM339集成电路是很常用的四电压比较器，它的外形接脚功能等如图 6-4 所

示。

在一块LM339的内部，有四个性能相同的比较电路，这里根据需要使用了其中的三个。

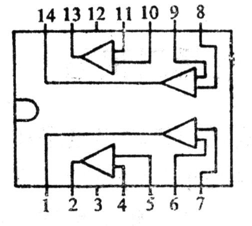


图 6-4　LM339接脚功能图

根据不同测试点上的电压，测试笔电路的三个发光二极管LED中有一个会发光。所测试到的电压首先被输入到两个比较器（A_1和A_2），每个比较器的另一输入端接一个基准电压，该电压由分压器R_4、R_5和R_6提供。对于TTL电路，所选择的门限禁区值在0.8V—2.4V之间。如果测试点TP的电压低于0.8V，A_2的输出端呈低逻辑并引起红色发光二极管D_6发光；如果被测点电压高于2.4V，A_1的输出呈低逻辑，则绿色发光二极管D_5发光。当然有时电压也会在0.8—2.4V之间，那么A_1与A_2的输出都不是低电平，当测试点TP悬空时，由于R_1和R_3的作用，使A_1和A_2的输出都是高电平，A_3的反相输入则通过R_9上升呈高电平，输出为低电平，因此黄色发光二极管D_7发光。

这里所用的元件无严格的规定，D_1—D_4也可采用国产2CP型二极管，IC_1可采用国产SG339等。只要元件合格、装置无误即能正常测试。装置的测量精度是否达到要求，可从测试点TP送入不同范围的电压来验证。

由于选用的元件体积较小，所有元件都可装入一线状狭窄的电路板上，然后套入透明管形外罩，这样可清楚地看到LED发光状态。

第三节 微处理器“听诊器”

随着微型计算机的普及，日益需要维修人员和设备。本节介绍的微处理器“听诊器”只使用了很少的器件，但却能够诊断出微型机的某些毛病。检查微处理器的工作状态时，常常需要准确地了解微处理机系统的地址、数据或控制总线上的逻辑。使用逻辑分析仪能方便地进行检查，但它的价格相当昂贵。我们自制的“听诊器”也能检测微处理机系统。当然，它也不是万能和最佳的测试设备，不过，即是医生的听诊器，也有一定的局限性。

“听诊器”的实际电路是非常简单的，如图6-5所示。

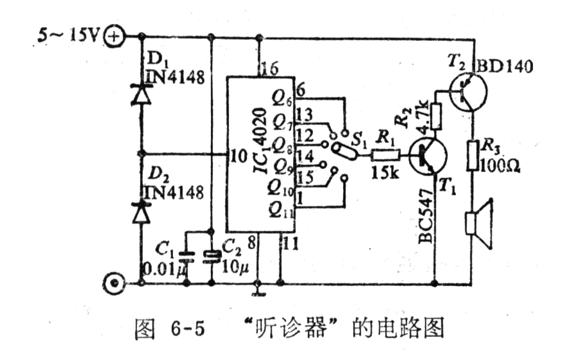


图 6-5 “听诊器”的电路图

“听诊器”可由待测试的电路供电，它的探头可用一般金属材料自制。探头接到分频器的时钟输入端。输入信号频率被预定的分频系数分频，其分频系数取决于选择开关S_1接通的IC_1的输出点。当S_1接到位置1时，分频输出为488.3 Hz/MHz。当S_1接到位置6时，分频输出为15625Hz/MHz。通过改变选择开关的位置，我们能将已变为可听频率的信号

折算成被分频的输入信号频率。假如我们想在几个测试点上跟踪1MHz时钟信号，若S_1在位置1，我们就可以听到一个约488Hz的信号。

时钟信号是一种常见的周期信号，如要周期信号出现在三条总线上，则微处理器必须工作在一循环程序上。这可加进部分监视程序，如果在进行例行测试时，无论按下任何一键即可。特殊的测试程序也可以用，并有上千种可能的程序，但取决于进行哪种测试。例如下面的6502程序可用于测试数据线和地址线。测试程序可直接存入（无须更改）到存储器的任何单元。

因为处理器在读周期性的操作码和运算数时，读/写信号将会周期性地出现，所以，若在微处理器的总线上要有确定周期的时钟信号，需要一些周期性指令，这时我们能在听诊器的输出信号中听出来。当然，这些输出信号是由S_1选定的分频系数经分频后得到的。

（附6502程序）

```
            CLC
LOOP        BCC   LOOP
```

IC_1各输出点接通S_1时的分频输出为：

$1 \div 2^{11}$	(2048)→	488.3 Hz/MHz
$2 \div 2^{10}$	(1024)→	976.6 Hz/MHz
$3 \div 2^{9}$	(512)→	1953.1 Hz/MHz
$4 \div 2^{8}$	(256)→	3906.25 Hz/MHz
$5 \div 2^{7}$	(128)→	7812.5 Hz/MHz
$6 \div 2^{6}$	(64)→	15625 Hz/MHz

第四节　微处理器状态显示器

在微型机的修理过程中，寻找微处理器系统的故障是一项艰难而又费时的工作。倘若能显示出微处理器数据总线上和地址总线上的信息，对寻找微处理器故障将是十分有益的，但这一点连普通示波器也难以完成，一般需要使用昂贵的逻辑分析仪。本节介绍的微处理器状态显示器电路并不复杂，却能完成上述工作，它的主要功能是，把数据信息和地址信息转换为十六进制代码送进六个显示器中去。图6-6是其原理图。

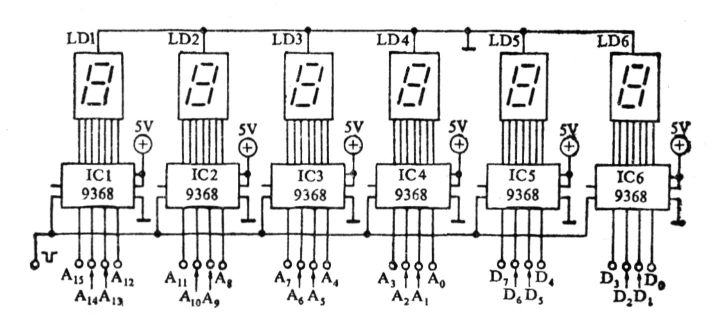


图 6-6　微处理器状态显示器电路

该电路由一种集成电路9368控制着六个显示器。9368*IC*具有电路所要求的全部功能，它接受一个四位二进制代码，并把它转换成相对应的十六进制数，并由一七段发光二极管显示器进行显示。电路也能提供自保持记忆功能并且直接由该*IC*块控制发光二极管进行显示。从图6-6中还可以看到，除了六个9368*IC*和六个显示组件外，电路没有其它的附加元件。由于六组电路的组成完全相同，所以该电路很容易阅

图。电路所需电源的电压为5V，电流约为750mA，主要供显示组件用。

在所有元件安装完后，应把该电路与微处理器相连。连接的导线应用带状电缆，长度适中，连接的原则是：前四个显示器显示地址，后两个显示器显示数据，外接电源的地线必须与微型机的地线连接在一起。

使用该电路的方法有两种：第一种是单步法，此法必须使 LE 端的电平为0V，这样，显示器就会显示出每步的数据和地址；第二种方法是由计算机本身控制 LE 端，以便能保持特殊数据（可能是暂时的）和地址，这种方法不需要停止计算机的工作，这是第二种方法的特点。

第七章　设备故障的分析和排除

第一节　APPLE-Ⅱ微型机的故障分析

我国APPLE-Ⅱ微型机的拥有量大、用户多，随着计算机应用的日益普遍，在使用中难免遇到一些机器故障，掌握一些故障维修技术，将会给你的工作带来很大的方便，本节很多实验或制作要用到APPLE-Ⅱ，因此，在实验或制作中也有可能遇到机器故障。本节将通过对两个常见故障的剖析，介绍一些常用的故障分析方法，达到举一反三的目的。实践证明，只有按照科学的思路，根据现象并借助仪器测量分析、逐步缩小查找故障的范围，才能找到故障点。在维修中切忌乱拔乱插、盲目查找，这样不但不能找出故障，甚至反而造成机器的损坏。

一、磁盘接口卡故障

1．故障现象：加电以后，不能引导磁盘，也不出现任何正常的提示，满屏幕都是无规则的字符或图形。

2．故障查找：

（1）首先确定是主机故障还是外设故障：拔下全部外设接口卡，若主机正常，则问题在外设或连接外设的接口卡。

（2）再确定是哪一台外设故障：逐个往回插入刚拔下

的接口卡，找出有故障的那一个。

（3）再确定是接口卡还是外设本身故障：用备用件交换比较。若没有备用件可交换，则检查接口卡槽的静态电平，然后从引脚信号倒推追踪进行分析。

根据以上分析，最后判定故障是由于软盘接口卡未进入工作状态所引起。

需要说明的是，以上操作要频繁地拔卡件。因此要注意两点：

其一，插拔接口卡一定要在断电情况下进行，而且关机之后要稍停一会儿再开机。

其二，关机之后不要急于触摸主机内部，最好先用手触摸一下主机裸露的金属部分（逻辑地），放掉人体静电。

3．故障分析：已知磁盘接口卡出故障，要进一步确定故障位置，就要从接口卡原理入手进行分析。

（1）接口卡的组成：一般软盘接口卡总是由数据转换电路、马达控制电路、驱动器选择电路和控制信号产生电路几部分组成。

数据转换电路具有串入并出，并入串出和左右移位的功能，所以一般由74LS323、D型触发器、ROM电路，外加一些与非门、非门电路片组成。

当接到马达启动信号时马达控制电路立即启动驱动器的旋转马达，当马达启动信号消失后，其输出信号要延长一段时间后再消失。这个电路一般由556之类的电路组成。

驱动器选择电路一般用与非门实现。控制信号产生电路由一个带锁存的多路器组成，一般用9334之类的电路。

根据组成，再对照实物，不难找到各个元件的位置和主要信号之间的关系。

（2）接口卡的工作过程：通电以后，主机从第七槽开始逐槽扫视看哪个槽插有什么卡。因此，首先查到的就是软盘接口卡。若有，就转去执行该接口卡上ROM的程序，从软盘上调进操作系统。对于较新式的APPLE机（如APPLE-Ⅱ PLUS），一部分操作系统已固化在ROM中，若不用软盘，就要从键盘上给一个信号（如APPLESOFT BASIC的CTRL—RESET），主机接到这个信号以后就从ROM中调出系统程序执行。为便于分析，我们把有关的一部分电路画出来，如图7-1所示。

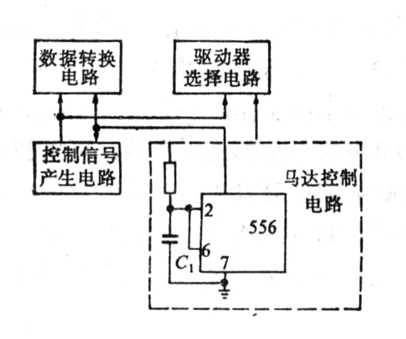


图 7-1 磁盘接口卡的组成框图

具体工作过程是这样的：加电后，556的RES信号变为高电平。这时该复位的都已复位，接口卡处于准备工作状态。当CPU执行“LDA $ CO89, X”指令（X是接口卡所在槽数乘以16）以后，就使接口卡进入工作状态，一号驱动器工作，旋转电动机开始转动。

如果C_1电容开路，则通电瞬间不能确保556时基电路的2、6端处于地电平，从而也就导致接口卡不能处于准备工作状态；如果C_1电容器短路，其危害更是显而易见的。因此，C_1这个电容很关键。取下C_1检查，果然开路了，换后即全部正常。

二、无闪烁的光标

1. 故障现象：开机后，主机的喇叭“嘟”的响了一声，过

了一会儿屏幕上出现了APPLE-Ⅱ的字样，并进入APPLE SOFT状态，出现了提示符“]”，但没有闪烁的光标。这时，主机与外设都能正常工作，调入的程序也都可以运行，键盘的字符输入也正常，但由于光标不能出现，使得机器进入编辑状态后，不能判断光标的位置，以致无法进行程序的编辑和正确修改。

2．故障分析：由于微型机电路较为复杂，集成电路多，所以必须采用逐步分析排除法，把故障范围限制在很小的局部电路中。由于此时微型机基本能工作，说明了主机的CPU电路、RAM电路以及ROM中的程序都是正常的，可以初步断定故障出在I/O电路或视频信号发生器的某一部分电路上。但从开机的工作情况看，I/O电路中的复位清零电路工作是正常的，这样就基本上可以确定故障是在视频信号发生器的电路中。

又由于键盘字符输入正常，所以故障电路应该是在闪烁/反相电路及闪烁定时器电路上。这两部分电路的框图如图7-2及图7-3所示。

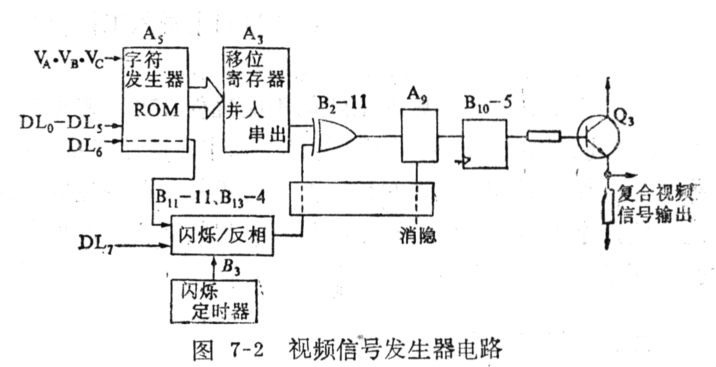


图 7-2 视频信号发生器电路

在图7-2中闪烁定时器B_3提供使字符和光标闪烁的定时

信号。门电路B_{11}和B_{13}对每个字符的高2位（DL_7和DL_6）译码以决定显示方式：字符闪烁是指字符的显示方式在正常显示方式和反相显示方式之间快速改变。

3．故障检查：

（1）检查图7-2中闪烁/反相集成电路B_{11}、B_{13}和B_3的电源是否加上，接地端是否接好，如果经检查都正常，那么，故障由电源引起的可能性可以排除。

图 7-3　闪烁定时器电路

（2）用示波器测量B_{11}集成电路的11、12脚及B_{13}集成电路的5、6脚，如都有正常的方波，但B_{11}的13脚并无方波出现，可初步判断B_{11}损坏或闪烁定时器B_3电路损坏。

（3）用万用表测B_{11}的13脚，如发现该脚电位为高电平，并无振荡现象（正常时该脚应有频率为4Hz方波），由于B_{11}的13脚始终有高电平存在，即相当于DL_6为1的情况，无法产生闪烁的显示方式。因此拔去B_3这块集成电路，用导线短路B_{11}的13脚接地，这时屏幕上有不闪烁的光标出现，说明故障发生在闪烁定时器电路上。

（4）更换一片好的B_3集成电路，如发现故障并未排除，B_3的输出端（B_{11}的13脚）仍为高电平，这可能是由于电阻3.3MΩ或0.1μF的电容损坏产生的，关掉主机电源，拔去B_3集成块，如测量电容器两端的电阻仅有数十kΩ，可焊下电容测量，发现电容漏电，更换一个电容后，主机即恢复正常。

第二节 APPLE-Ⅱ复位电路故障的分析与排除

一、故障现象

开机后，显示器上出现一整屏幕方块，磁盘驱动器不工作，按任何键均无作用。

上述故障现象，大都是由于送给 CPU 第 40 脚的系统复位信号RES恒为低电平引起的。

二、系统复位原理

标准的复位信号应是一个先低后高的电位信号。从所给的电路中可以看出，复位信号有三个来源：

1. 主板上的复位电路：如图7-4所示，主板上的复位电路是由时基电路A_{13}（型号NE555）、电阻R_{26}、R_{14}、电容C_4和晶体管Q_5等组成。

开机时，V_{CC}经R_{26}向C_4充电，V_{TR}（即C_4两端电压）$<(2/3)V_{CC}$时，A_{13}的输出端$V_{OUT}=1$。此高电平经Q_5反相变为低电平后加到CPU的复位端(RESET端)，因而RES＝0 。

由于C_4不断被充电，约 1 秒钟（由R_{26}，C_4之值决定）后，$V_{TR}\geqslant(2/3)V_{CC}$，使得$A_{13}$内部触发器翻转，故$V_{OUT}=0$。同理，经$Q_5$反相，RES＝1。这样，便产生了一个完整的RES信号。

2. 磁盘控制板上的复位电路：磁盘控制板上的复位电路与主板上的复位电路基本相同，唯一的区别是决定复位时间（即开机后RES＝0的时间）的电阻R_{26}与R_4之值不用，故

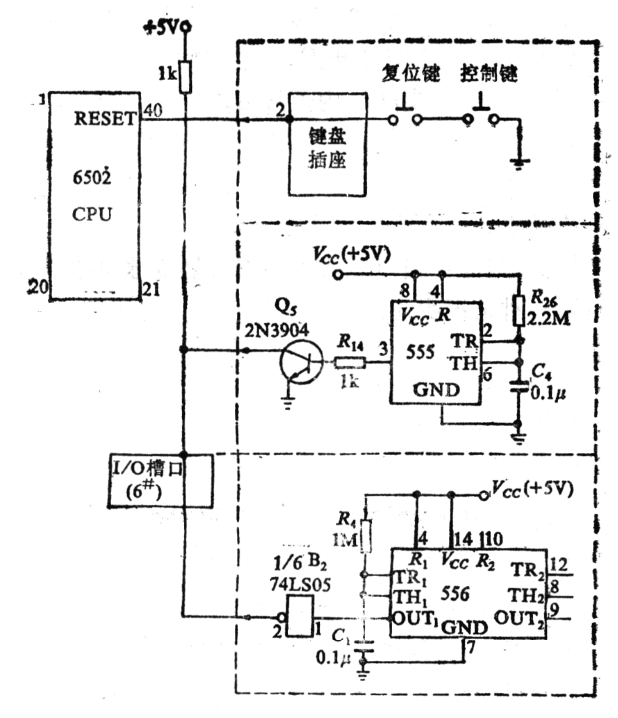


图 7-4　主机板上的复位电路

其原理不再赘述。

3．键盘：如图7-4所示：复位键与控制键组成一串联开关，两个键同时按下、放开一次，对中央处理机的复位端而言，相当于给了一个先低后高的复位信号。

三、故障原因

1．主板（或磁盘控制板）复位电路异常：

（1）C_4（或C_1）漏电较大，性能变坏，使得A_{13}（或D_2）

的TR端电压V_{TR}达不到(2/3)V_{cc}，因而输出端V_{OUT}恒为高电平，使CPU一直处于复位状态。

（2）由于A_{13}（或D_2）内部故障，使得输出端V_{OUT}恒为高电平。

（3）由于反相器Q_5（或(1/6)B_2）的输出对地电阻变小，使中央处理机一直处于复位状态。

2．键盘有问题：当电缆线短路或者复位键和控制键触点短接（后一种情况少见）时，也将使中央处理机一直处于复位状态。

四、故障排除

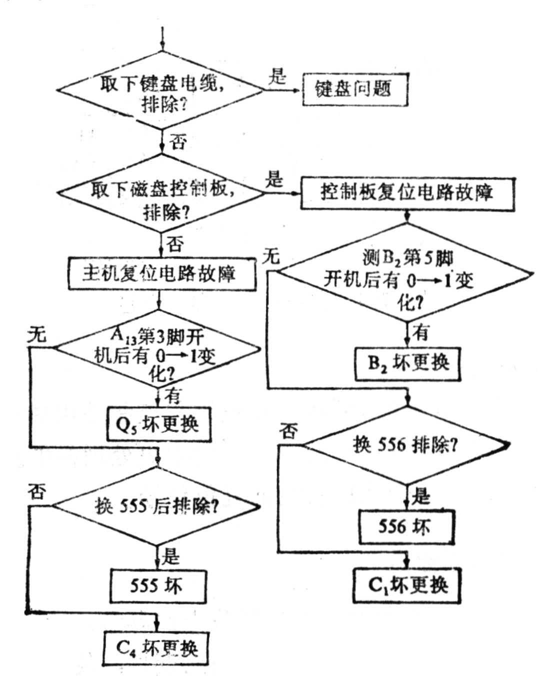


图 7-5　复位电路故障的分析与排除方法

当出现本节开头提到的故障现象时，可先用逻辑笔测试中央处理机第40脚之状态。若一直为低电平，则可按图7-5给出的框图，去查找故障部位及器件。

注意：插、拔电路板及集成电路芯片时，必须在关机的情况下进行。在用电烙铁烙元件时，电烙铁外壳保证接地，或把电烙铁的电源插头拔掉。

第三节 APPLE-Ⅱ常见集成电路故障

微型机损坏的原因一般有两类：一类是自然损坏，另一类是用户使用不当所引起的损坏，其中后一类损坏，特别是用户使用不当所引起的集成电路损坏为数不少。因此，正确地使用计算机，可减少或避免计算机故障。

要注意的是，使用微型机前应先检查主机和显示器（CRT）的电源线是否接上，主机的视频输出线与显示器是否联上，以及所需使用的扩展卡是否已插入扩展槽内。开机时，应先打开显示器，然后打开主机。开机后各扩展卡和集成块不得再从主机板上拔出或插入。否则就可能使主机电源或集成块损坏，产生故障。插换集成块时，必须把集成块的每个引脚都插入插座内，并注意集成块引脚的排序，切不可漏插或插反。在微型机已发生故障不能正常使用时，应查明故障原因，排除故障后再使用。下面介绍由APPLE-Ⅱ微型机元器件损坏引起的故障及其现象和检查方法。

一、晶体损坏：晶体用于产生14.318MHz的主振频率，通过14分频得到1MHz的时钟脉冲，送入中央处理机(6502)第37脚。晶体损坏后，基准振荡系统就不能产生主振脉冲，中央处理机就无法工作。用示波器可以发现6502第37脚无时

钟脉冲，同时B_2集成块74S86第8脚，B_{11}74LS68第3脚均无脉冲信号。所表现的故障现象是开机后，喇叭不响，屏幕无任何显示。

二、B_{11}与门74LS08损坏：6502正常工作时，读写端$R/\overline{W}$通过8T97缓冲器接到或门C_{14}与ϕ_1（$\phi_1=\overline{\phi_0}$）一起控制数据缓冲器74LS245，而$R/\overline{W}$由$\overline{DMA}$（数据直接存取线）通过C_{11}非门控制（见图7-6）。

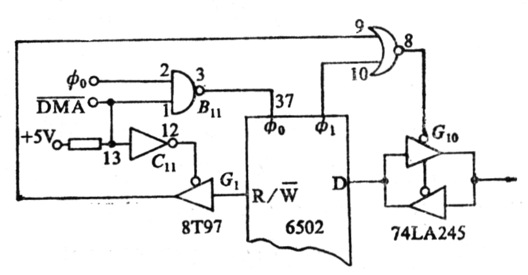


图 7-6　查找B_{11}故障的有关电路

一般情况下，$\overline{DMA}$始终为高电平，8T97始终打开，中央处理机正常工作。当使用Z-80、8088、6809扩展卡时$\overline{DMA}$处于低电平，使6502停止工作，扩展卡上的中央处理机工作。因$\overline{DMA}$在6502工作时始终为高电平，故在6502的第37脚就得到了1MHz的时钟脉冲。但是当B_{11}损坏以后，使其原为高电平的第1脚被箝位在低电平。通过与门使得6502的第37脚亦为低电平，6502停止工作。用示波器可以发现B_{11}的第2脚有1MHz脉冲信号，第1、3脚为低电平。故障现象为主机不工作，喇叭无声、屏幕无显示。

三、与或非门74LS51（C_{13}）损坏：在图7-7的同步电路中，C_{13}的输出用于产生合成同步信号（帧、行同步信号），使显示器产生稳定的字符和图象显示。C_{13}的输入端分别接字

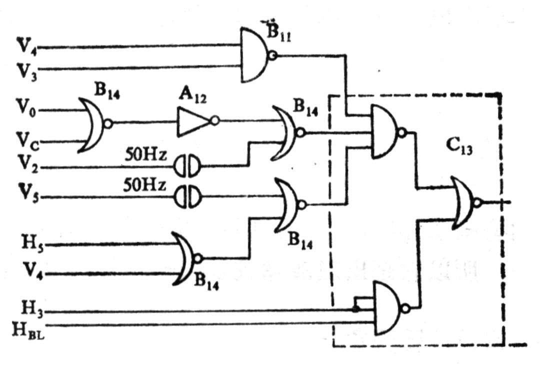


图 7-7 查找C_{13}故障的有关电路

符列计数器、字符行计数器、扫描线计数器的输出端，通过C_{13}后得到和显示器同步的脉冲信号。由于C_{13}损坏，使得主机不能产生稳定的同步信号。故障现象是开机后系统能正常工作，但屏幕显示产生扭曲和翻滚，调节帧、行频旋钮无法控制。

四、74LS02或非门（B_{13}）损坏：B_{13}属于文本、图象产生电路，用于文本状态和图象状态之间的转换。由于B_{13}的损坏，使主机不能进入作图状态（打入GR或HGR命令均无法进入作图状态）。故障现象是打入GR命令后，出现的图象同打入TEXT命令后一样。

五、数据缓冲器74LS245（G_{10}）损坏：G_{10}是中央处理机位数据线上的三态门（参见图 7-6），通过读写信号 $R/\overline{W}$和ϕ_1脉冲控制，中央处理机不断读入数据或对 RAM 进行写入操作。74LS245根据控制端的电平高低来确定是否打开某方向上的数据三态门。G_{10}的损坏使得中央处理机不能产生$R/\overline{W}$脉冲及地址信号，不能找到正确的RAM地址。用示波器检查，可以发现$R/\overline{W}$端始终为高电平，所有地址线A_0～

A_{15}均处于高电平或低电平状态。故障现象是开机后主机不工作。

第四节　打印机常见故障的排除

在微型机系统中，由于打印机的机械运动部件多，并且使用频繁，所以它是出现故障较多的一种外围设备。

一、常见故障现象

打印机最常见的故障多发生在电源部分及驱动电路部分，故障现象常表现为：

1．接通电源后烧保险丝，或报警（说明电路内部有严重短路）。

2．打印头不动，或不能走纸（常为驱动电路部分发生故障，即步进电机及其驱动线路故障）。

二、检修

1．电源部分的检修：首先观察电源变压器有无过载发热现象，判断变压器内部是否有短路，测量保险丝是否开断。然后用数字万用表的"—|◁—"档（或指针式万用表的Ω档），在线测量整流桥，桥臂上的每个二极管的正反向都要测一下，正向压降为0.2～0.7V，只要正反向读数差别很大，就可以认为整流桥正常（不必将它拆下）。还应测量稳压调整管是否损坏，可分别测量b—e，b—c的正反向压降和e—c间阻值，观察是否有击穿或者开路以及压降是否为正常值。如用晶体管特性图示仪测量那就更方便了。

最后，再用数字万用表的"—|◁—"档测量直流电源的

正负两端（不能接通电源）。红表笔接+24V测量点，黑表笔接地端，这时从表上数字的变化可看到有充电现象，否则，多为滤波电容击穿。

在以上检测中，如果发现元件损坏，可用同型号或相近参数的元件加以代换。若在测量中，尚未判断出故障时，不要急于接通电源，最好继续往下检查，否则，因为短路因素并未排除，仍会烧坏元件，造成损失。

2．驱动线路的检修：驱动电路属于强电流回路，若某处短路，表现为打印头不动或不能走纸，同时报警，严重的还会烧保险丝，故对此部分线路应仔细检查。

步进电机的驱动电路一般由开关管等组成，首先要判断每一个开关管是否完好，若发现某一路损坏，则可将三极管调换，用数字万用表的“—|◁—”档在线测量，不必将元件焊下，只要正反向压降差别大，正向压降正常，就可判断该管完好。注意在测量这部分线路时，应将步进电机与线路板断开。

经过上述检查后，一般可找出驱动电路中的强电流短路的故障点，进而排除故障。较为典型的驱动电路如图 7-8 所示。

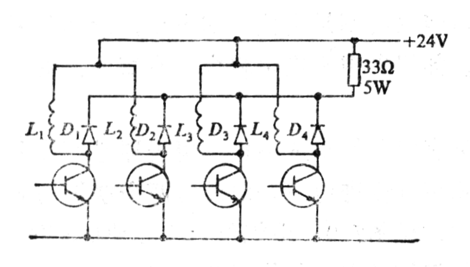


图 7-8　步进电机驱动电路

3．步进电机的检修：步进电机内部短路现象常有发生，特别是走纸步进电机，由于纸被卡住等原因，造成电机长时间过载发热，烧坏漆包线漆层，造成线间短路或匝间短路。步进电机内部短路时，若不及时关机，还会烧坏驱动管或电源部分元件，使故障扩大，因此排除这部分故障也非常重要。

检查时应先拔下步进电机插头，用数字式万用表的220Ω电阻档分别测量各线包的直流电阻。以走纸步进电机为例，其中引线1—5和3—5以及2—6、4—6的直流电阻一般为28Ω；1—3的阻值为1—5、5—3阻值之和，2—4为2—6、6—4阻值之和，其阻值一般为56Ω。如图7-9所示，图中所标为某电机的实测值。对于不同的电机，实际阻值略有差别。

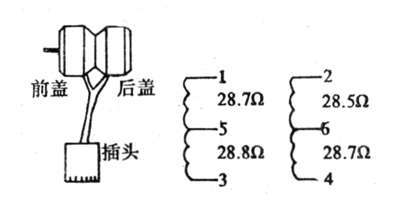


图 7-9　步进电机的线圈电阻

对比之下，如果某相阻值偏离太大（大于10%）可判断该相有问题。打印头步进的直流电阻值较大，1—3、2—4的阻值为66Ω，5，6分别为各组中心抽头。实践证明，用测量直流电阻（而不是电抗）的方法，能够判断出步进电机是否短路。如果发现步进电机内部有短路，可用以下方法修复。

拆下有问题的步进电机，用表测量，确定是前部还是后部线包损坏（步进电机是由前后两个绕组构成，每个绕组有两组线圈，见图7-9。用螺丝刀小心地撬下欲修部分的端盖，取出线包仔细检查，如能找到短路处，拨开短路点，涂上绝缘胶或绝缘漆，就可修复，若故障严重，必须重绕，重绕的

方法如下：

（1）拆掉已损坏的线包，同时准确地记下绕线的方向及圈数，量出所使用的线径（本例中使用0.2mm高强度漆包线）。

（2）用同线径的漆包线，双线并绕，绕至与原线包的圈数相同。应注意要绕得平整，同时要与原线包的绕向相同。绕好后的线包，其填充程度，直流电阻均应与原绕组相近（前后两绕组的直流电阻也应相近）。

（3）涂绝缘胶（如702胶）、或浸漆，烘干。

（4）接上引线，用绝缘胶带在线包外包两层，装入电机，盖好端盖，电机线包就绕好了。

经过以上步骤的检修，对于因电源、驱动电路部分所引起的打印机故障一般均可排除。

三、其它故障

1．突然出现打印机乱打现象（即属于逻辑控制部分的故障）。在实际使用中，出现这类故障的机率较少。可通过更换控制板，或先判断出产生故障的IC模块后再加以更换，使故障得到排除。

2．走纸不准：如果走纸不准的现象只出现在第一行，那么在需要走空纸时，应利用LF键机械走纸，注意不要在打印前手动走纸。

如果走纸不准的现象是间断地出现的，那么可检查纸牵引器传动部件的齿轮间有无积尘异物，齿轮是否磨损或破裂。若是后者，则需更换齿轮。如果以上两项措施均无效，则需检查驱动牵引器电机的输入电压是否正常。

3．打印字迹不清：可采取以下措施：

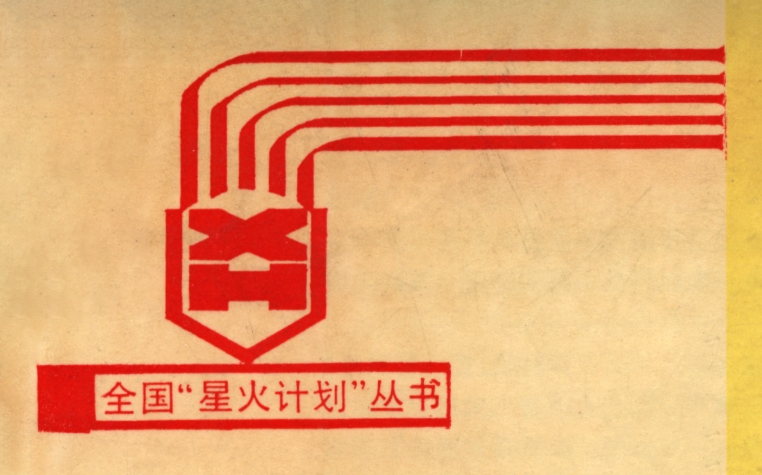
（1）调整打印头间隙；

（2）更换色带；

（3）清洗打印头。具体办法是：将固定打印头的螺丝松开，然后将打印机倾斜，用棉花蘸少许汽油或无水酒精擦洗针头。

4．打印头滑动呆滞：当打印机使用较长时间后，打印头传动波带就会出现宽松现象，使打印头滑动呆滞。这时可将左部打印头驱动部件的螺丝松开，使打印头驱动部件向左移动，以接紧皮带，然后固定。调整后，来回移动打印头，使它能在机轴上顺利平滑移动即可。





青少年计算机学习与应用丛书部分书目

1. 中华学习机 CEC-I 技术参考手册（硬件）
2. 中华学习机 CEC-I 技术参考手册（软件）
3. 中华学习机小蜜蜂-I 使用与技术参考手册
4. 电脑天地
5. 苹果机和中华学习机应用实验与实用制作
6. 青少年计算机获奖教学软件选编

（以上软件均可在中华学习机、苹果机上运行。）

ISBN 7-302-00458-7

TP·158　定价：2.80元

苹果机和中华学习机应用实验与实用制作

清华大学出版社